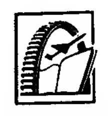
НАУЧНО-ПОПУЛЯРНАЯ БИБЛИОТЕКА ШКОЛЬНИКА

В.В. АЛЕКСАНДРОВ, В.С. ШНЕЙДЕРОВ

Рисунок чертеж картина на ЭВМ



Ленинград
"Машиностроение"
Ленинградское отделение
1987

ББК 32.97
А46
УДК 681.3.004.14

Рецензент канд. техн. наук И. А. Румянцев

Александров В. В., Шнейдеров В. С.
А46 Рисунок, чертеж, картина на ЭВМ.— Л.: Машиностроение. Ленингр. отд-ние, 1988.— 128 с., ил.— (Научно-популярная библиотека школьника)

В книге в популярной форме рассказано об электронно-вычислительных машинах, которые умеют чертить, рисовать и генерировать цветные изображения. Кратко описано их устройство. Показано, как можно дать задание ЭВМ и использовать ее в качестве инструмента конструктора, проектировщика, инженера и специалистов других профилей. Приведены примеры использования "электронных художников" на производстве, в архитектуре, в искусстве. Изложение построено на базе материала программы средней школы.

Книга предназначена для школьников, учащихся ПТУ и техникумов.

А $\frac{2405000000-960}{038(01)-88}$ 171—87

ББК 32.97

ПРЕДИСЛОВИЕ

Вычислительная машина — инструмент новой информационной технологии, порожденный постоянно прогрессирующей электронной технологией. Из многочисленных областей применения компьютера нас, в первую очередь, интересует использование его как инструмента чертежника, конструктора, художника, режиссера, что позволяет освободить представителей этих профессий от неблагодарной рутинной работы. Общим для этих четырех профессий является работа с образной информацией.

В Японии большое число часов в школьной программе отводится рисованию, так как написание иероглифов требует очень большого искусства. Алфавитный принцип, конечно, намного упрощает представление текстовой информации. Но в соответствии с законом сохранения энергии за такой способ помехозащищенности приходится расплачиваться как неоднозначностью смысла слов, так и медленным развитием содержания.

В этом отношении образная информация выгодно отличается от текстовой. Приведем два высказывания. "Я думаю, что чертежи очень полезное средство против неопределенности слов" (Лейбниц, 1646–1716 г.). "Своеобразие геометрии, выделяющее ее из других разделов математики, да и науки вообще, заключается в неразрывном, органическом соединении живого воображения со строгой логикой. В своей сущности и основе геометрия и есть пространственное воображение, пронизанное и организованное строгой логикой. В ней всегда присутствуют эти два неразрывно связанных элемента: наглядная картина и точная формулировка, строгий логический вывод. Там, где нет одной из этих сторон, нет и подлинной геометрии"[1].

[1] Из предисловия акад. А.Д. Александрова к книге К.Е. Левитина "Геометрическая рапсодия" (М., 1984).

Необходимость изучать начертательную геометрию и проекционное черчение, выполнять трудоемкую и кропотливую работу чертежника отбивает охоту к образному выражению своих идей у большинства людей. Компьютер же легко справляется с автоматизацией процесса геометрическо-логического построения, т. е. берет на себя часть левополушарных функций мозга (как утверждают ученые, предрасположенность человека к абстрактно-логическому мышлению предопределяется преимущественным развитием левого полушария головного мозга).

Кроме того, чаще всего разработка нового проекта, конструкции связана с модификацией отдельных элементов и узлов. И здесь компьютер выступает одновременно как архив проектов и стандартов, и как устройство, тиражирующее конструкторско-технологическую документацию, что позволяет резко повысить эффективность труда.

Дизайнеру этот инструмент позволит более успешно вести поиск новых форм, разрабатывать удачные находки предшественников, например Александра Родченко, Виктора Вазарели, Моурица Эсхера. Метод аппликации, или коллажа, предложенный Родченко еще в 1919 г. для оформления журналов, афиш, рекламных изданий, отпугивал трудоемкостью выполнения подобных композиций. Это препятствие полностью снимает компьютер, позволяя создавать "электронные аппликации" (см. гл. 7), так же как и проблему поиска цветовых декоративных сочетаний, разбиения плоскости картины и заполнения ее симметричными, асимметричными и рекурсивно повторяющимися элементами в духе Эсхера.

ВВЕДЕНИЕ В МИР ИЗОБРАЖЕНИЙ И МАШИННУЮ ГРАФИКУ

Близ испанской деревни Альтамира сохранилась пещера — жилище первобытного человека. Скалистые выступы на потолке очерчены линиями и покрыты краской. Это изображение бизонов. Один щиплет траву, другой лежит, третий падает от стрелы охотника.

Эти ранние формы графики были первыми попытками передачи информации об окружающем мире последующим поколениям. Они представляли собой своеобразный учебник жизни.

Люди начали рисовать уже в те незапамятные времена, когда жили в пещерах, добывали огонь трением, шли на охоту с каменными топорами, т. е. задолго до того, как научились писать. В Сибири, в Кузнечном Алатау найден рисунок, возраст которого — 34 тысячи лет! Наскальные росписи выполнялись земляными красками (красными, желтыми и коричневыми охрами, умброй), черной сажей и древесным углем с помощью расщепленных палочек, кусочков меха и просто пальцев.

С тех пор прошли многие тысячи лет, появились письменность и книгопечатание, человек овладел энергией атомного ядра и вышел в космическое пространство, а что изменилось в технике рисования? Стали лучше краски, кисти, появились перья, карандаши, фломастеры, но в принципе все осталось тем же самым: та же цепочка глаз — рука — инструмент — изображение, те же требования к способностям художника.

Но вот появилась вычислительная техника. Вызванная к жизни необходимостью автоматизации решения трудоемких математических задач, ЭВМ из большого калькулятора неожиданно превратилась в интеллектуальный инструмент, сфера приложения которого стремительно расширяется. В начале 1960-х годов родилась новая область вычислительной техники — инте-

рактивная[1] машинная графика, где компьютер используется уже не столько для обработки чисел, сколько для работы с графической информацией.

Интерактивная машинная графика — это, образно говоря, интеллектуальная кисть, т. е. кисть, наделенная знанием. Благодаря этому кардинально изменяется процедура создания графической информации. Из инструмента, повторяющего механические движения руки, она превращается в инструмент, управляемый человеческим воображением. В то же время за счет накопленного знания машинная графика отграничивает область разрешенного (возможного) от пустых фантазий. Машинная графика в значительной степени определяет эффективность внедрения автоматизированных систем в сферы проектирования и производства.

Техническое и программное обеспечение компьютеров достигло такого уровня, когда человек уже может не подстраиваться под компьютер, наоборот — компьютер подстраивается под человека. Машинная графика обеспечивает наибольшую "гуманизацию" вычислительной техники, упрощая процедуры общения пользователя с ЭВМ, обеспечивая наглядное отображение информации, максимально раскрывая творческие возможности человека.

Есть такие задачи, результат решения которых просто невозможно воспринять без графического вывода, например поведение автомобиля в аварийных ситуациях, вид на аэродром с самолета, строение молекул и т. п.

В названии нашей книги выделены три вида изображений: рисунок, чертеж, картина. Это вызвано тем, что каждый из них имеет свою специфику, что проявляется в особенностях устройств ввода, хранения, обработки и отображения графической информации. Под рисунком мы понимаем графическую форму изображения, в основе которой лежит линия, это контурное изображение. Чертеж является проекцией некоторых реально существующих или воображаемых объектов. Под картиной мы подразумеваем тоновое чернобелое или цветное изображение.

---

[1] *Интерактивные* устройства и системы рассчитаны на непосредственное взаимодействие пользователя с компьютером, когда на каждую команду пользователя следует реакция (ответ) компьютера.

Графические возможности компьютера не могут не вызывать изумления. В нашей книге мы попробуем снять покров таинственности с компьютера, покажем этапы его становления как интеллектуального помощника человека.

1. АНАТОМИЯ КОМПЬЮТЕРА

Чтобы рисовать, нужно знать особенности инструмента: кистью рисуют не так, как фломастером. Чтобы рисовать с помощью компьютера, нужно иметь представление о возможностях и особенностях машинной графики.

Все графические системы — и дорогостоящие, применяемые в авиационной и космической промышленности, отличающиеся высокими точностью и скоростью обработки информации, и дешевые персональные для решения простых графических задач или игр — содержат устройства: ввода графической информации, ее хранения (оперативная и внешняя память), обработки (процессор) и вывода.

Устройства вывода графической информации служат для отображения (графические дисплеи — векторные и растровые) и регистрации (печатающие устройства, графопостроители, устройства вывода на фото и кинопленку). Обычно устройства ввода, вывода информации и внешней памяти называют *периферийными.*

Информация об изображении, которое в настоящий момент обрабатывается процессором, временно хранится в оперативном запоминающем устройстве. В него же загружается программа обработки. Архив изображений, которые не нужны в настоящий момент, но скоро могут понадобиться, хранится во внешнем запоминающем устройстве. Изображение может передаваться из внешней памяти в оперативную и обратно. Наиболее распространенная конфигурация персональной графической системы приведена на рис. 1. Такую систему можно подключить к информационно-вычислительной сети большого вычислительного комплекса, тогда она станет *терминалом,* т. е. оконечным устройством для ввода и вывода информации. За счет подключения увеличиваются ресурсы машинной памяти и ускоряется обработка данных.

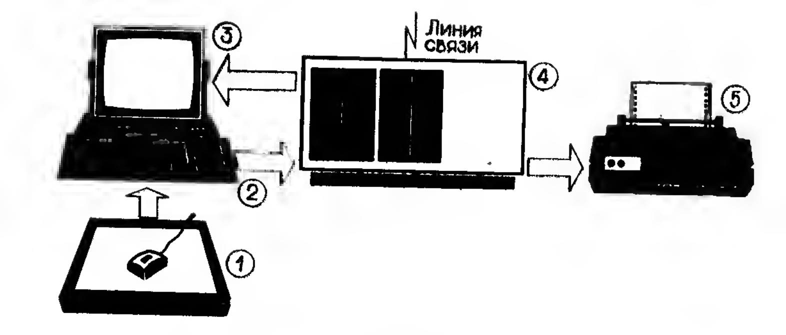


Рис. 1. Персональная графическая система:
1 — устройство ввода; *2* — клавиатура; *3* — дисплей (здесь же располагается и процессор); *4* — устройство внешней памяти (накопитель на магнитных сменных дисках); *5* — устройство регистрации информации

Устройства ввода. В машинной графике изображения делят на контурные, тоновые и цветные. В основе *контурных* лежит линия (рисунок, чертеж), в основе *тоновых* и *цветных* — поверхности, пятна (такие изображения мы будем условно называть "картинами"). Большинство устройств ввода графической информации предназначено для ввода рисунков и чертежей.

На рис. 2 показаны основные устройства ввода графической информации: световое перо, планшет, "мышь", рычаг, шар, сканирующая система (телевизионная камера). Все эти устройства, кроме сканирующих, предназначены для ручного ввода контурной информации. Они позволяют создавать рисунки непосредственно на экране дисплея и одновременно вводить их в компьютер.

Рассмотрим принцип действия перечисленных периферийных устройств.

Планшет. Это устройство представляет собой плоскую, обычно прямоугольную поверхность, на которую накладывают готовый рисунок и обводят его указкой. В процессе обводки компьютер периодически считывает позицию кончика указки. Для определения его координат существует ряд методов. Рассмотрим два из них.

Под рабочей поверхностью планшета размером, например, 250×250 мм вдоль осей x и y размещается по 1024 параллельных проводника. Каждый из них выполнен из медной полоски шириной 75 мкм и толщи-

ной 25 мкм. Расстояние между осями проводников равно 250 мкм. Группы проводников разделены тонкой пленкой. К каждому проводнику подводится определенный двоично-кодированный сигнал, который воспринимается указкой (рис. 3, *а*) через емкостную связь. В указке расположен чувствительный усилитель, который принимает кодированный сигнал и по кабелю передает его в декодирующую логическую схему. Каждая принятая последовательность импульсов соответствует только одному проводнику, поэтому после декодирования можно легко получить двоичное число, соответствующее координате положения этого проводника на поверхности планшета. Разрешающая способность планшета достигает 0,25 мм, т. е. графические элементы, меньшие 0,25 мм, не считываются.

В конструкции, изображенной на рис. 3, *б*, использован акустический принцип определения координат указки. На конце указки смонтирован искровой датчик, состоящий из двух электродов, разделенных керамическим изолятором, между которыми через регулярные промежутки времени проскакивает искра. Полосковые микрофоны, расположенные по краям планшета, воспринимают звуковые импульсы, а два счетчика отмечают интервалы времени между возбуждением искры и принятым звуковым импульсом. Значения времени задержки двух импульсов пропорциональны расстояниям от конца указки до двух сто-

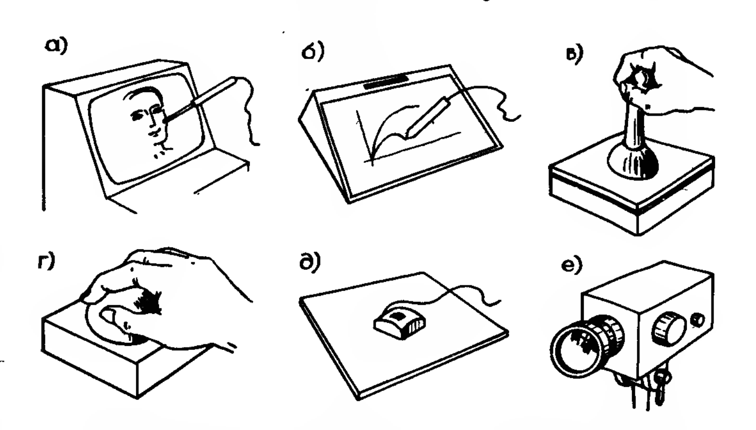


Рис. 2. Устройства ввода графической информации:
а — световое перо; *б* — планшет; *в* — координатный рычаг; *г* — шар; *д* — мышь; *е* — сканирующая система

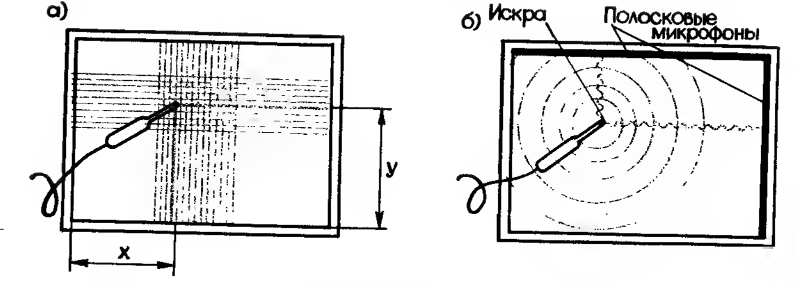


Рис. 3. Планшет:
а — на основе ортогональной решетки; *б* — на основе акустического принципа

рон планшета, на которых установлены микрофоны, поэтому их значения могут быть приняты в качестве координат x и y положения указки.

Световое перо. Световым пером рисуют непосредственно на экране. В корпусе пера имеется отверстие, в котором находится линза. Свет с экрана попадает на фотоэлемент [непосредственно (рис. 4, *а*) или через волоконный световод (рис. 4, *б*) — кварцевое волокно, по которому свет распространяется за счет полного внутреннего отражения от стенок волокна]. Напряжение с фотоэлемента после усиления подается на специальное переключательное устройство — триггер Шмитта[1]. Когда яркость света превышает некоторое пороговое значение, триггер вырабатывает импульс, используемый как сигнал для начала вычисления координат пера. Применение интегральных схем позволяет уместить усилитель в корпусе светового пера, который имеет размеры шариковой ручки. Световое перо снабжается переключателем, который пользователь приводит в действие, нажимая на кнопку или прижимая головку светового пера к поверхности экрана.

Рисование световым пером осуществляется способом, называемым "*слежением за пером*". На экран выводится специальный символ — *курсор*, имеющий вид креста. Курсор "прикрепляют" к световому перу, указав на него и нажав переключатель пера. Теперь куда бы мы ни перемещали световое перо, за ним, как привязанный, неотступно следует курсор. После

[1] Слово "*триггер*" происходит от англ. *trigger* (пусковое устройство).

вывода курсора в требуемое положение переключатель отпускают, и курсор "отцепляется" от пера. Теперь его центр указывает координаты точки. Две точки соединяются отрезком прямой.

"Мышь". Внешне это устройство ввода графической информации чем-то напоминает мышку, поэтому разработчики, не утруждая себя придумыванием точного технического термина, окрестили его "мышью". Мышь — это коробочка, легко умещающаяся в руке, с кнопкой наверху. На ее дне располагаются два взаимно перпендикулярных колесика. Колесики связаны с резисторами. Чтобы рисовать мышью, достаточно катать ее по ровной поверхности. При этом смотреть нужно не на мышь, а на экран дисплея, где траектория движения курсора повторяет траекторию движения линии. Это обеспечивается тем, что при вращении колесиков вращаются движки переменных резисторов и соответственно изменяются напряжения на их выходах. Снимаемые с движков напряжения обрабатываются аналого-цифровыми преобразователями для передачи непосредственно в оперативную память ЭВМ. Одно колесико смещает курсор по оси абсцисс, другое — по оси ординат. Катая мышь по поверхности, мы наблюдаем за движением курсора по экрану и соответствующим образом корректируем свои движения. После того, как мы добиваемся нужной траектории курсора, можно нажать кнопку и координаты курсора запоминаются компьютером. Обычно опрос координат производится с низкой частотой (от 30 до 60 раз в секунду), что позволяет использовать дешевые и простые модели аналого-цифровых преобразователей. Мышь обладает рядом интересных свойств: например, можно перенести ее на другое место, а курсор на экране остается не-

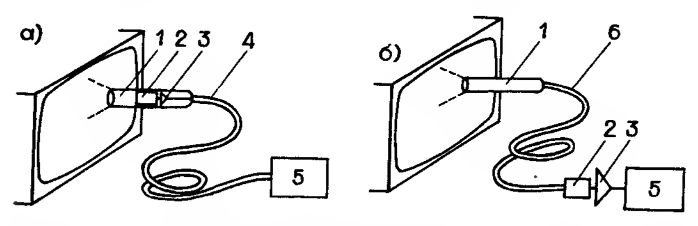


Рис. 4. Световое перо:
а — вариант с кабелем; *б* — вариант с волоконным световодом; *1* — линза; *2* — фотоэлемент; *3* — усилитель; *4* — кабель; *5* — триггер Шмитта; *6* — волоконный световод

подвижным. Если курсор доведен до края экрана, а мышь продолжает движение, то курсор перескакивает на противоположную сторону экрана и продолжает перемещаться в том же направлении. А вот ввести чертеж в память ЭВМ с помощью мыши мы не сможем, так как из-за проскальзывания колесиков при перемещении мыши происходит накопление ошибок.

Рычаг. У этого устройства несколько названий: джойстик (от англ. *joy stick* — палочка радости), координатная ручка, координатный рычаг. Мы будем называть его рычагом.

Рычаг имеет две степени свободы, т. е. может перемещаться в направлении вперед—назад и вправо—влево. Прикладывая минимальное физическое усилие, можно изменять наклон (угловое положение рычага) и таким образом перемещать курсор по экрану дисплея. Движения курсора довольно резки, что препятствует быстрой и точной установке курсора в заданное положение. Вследствие этого рычаг часто используется для управления скоростью движения курсора, а не его позицией. При этом текущая позиция экранного курсора изменяется со скоростью, определяемой положением рычага. Аналогичный принцип действия имеет устройство под названием "шар".

Каковы же принципиальные различия между рассмотренными выше устройствами? В устройствах ввода типа "планшет" и "мышь" пространственные перемещения руки преобразуются в перемещения курсора по *линейным* законам. Поэтому изображение, накатанное мышью или обведенное указкой, идентично полученному на экране. Пользуясь рычагом или шаром, пользователь задает угловые смещения, которые преобразуются в перемещения курсора по *нелинейным* зависимостям. Это необходимо для увеличения чувствительности устройства. В такой ситуации кинестетическая (мышечная) память, приобретенная человеком в ходе практической деятельности, вызывает ошибки. Но пользуясь обратной визуальной связью — наблюдая за перемещениями курсора — можно корректировать свои движения.

Изображение, введенное с помощью рассмотренных выше устройств, выглядит, как рис. 5.

Сканирующие системы. Сканирующие системы позволяют вводить в ЭВМ тоновые и цветные изображения. *Сканер изображений* (рис. 6) представляет собой

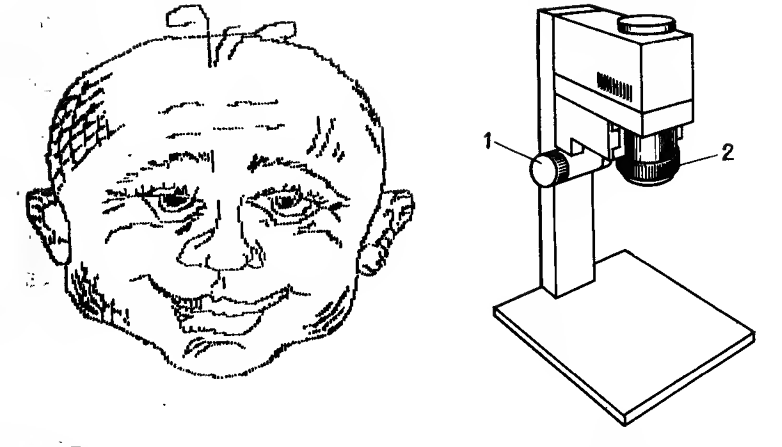


Рис. 5. Набросок, полученный с помощью светового пера

Рис. 6. Сканер для считывания изображений (репродукций): *1* — **регулятор высоты;** *2* — **кольцо настройки на фокус**

настольный прибор, используемый для ввода изображений с репродукций в графическую систему. Через объектив проецируется визирная сетка, позволяющая правильно разместить картинку и настроить объектив на резкость. Пользователь легко может направить камеру на нужный участок картинки. Входное изображение проецируется на линейную матрицу светочувствительных элементов размером 1728×1. Напряжения с каждого элемента матрицы преобразуются в числа и поступают в память ЭВМ. Матрица механическим способом плавно перемещается, пересекая плоскость изображения примерно за 12 с. При этом в памяти ЭВМ остается числовой образ двумерного изображения.

Сканером может быть и обычная телевизионная камера. Но чтобы с ее помощью ввести изображение в память компьютера, требуется высокочастотный аналого-цифровой преобразователь, который переводит значения интенсивностей каждого из трех цветов изображения (красного, зеленого и синего) в числа. С помощью телевизионной камеры вы можете ввести в ЭВМ практически любое изображение, например улицы или человека (рис. 7).

Обычная телевизионная камера не приспособлена для работы с компьютером. В настоящее время начинают появляться устройства, специально предназначен-



Рис. 7. Компьютерная "фотография" (две половины изображения различаются числом бит, приходящимся на 1 элемент)

ные для ввода в ЭВМ более дешевые и компактные. Так, имеются устройства, обеспечивающие дискретизацию изображения при вводе на 128×256 элементов.

Графические дисплеи. Графический дисплей (от англ. *display* — показывать, воспроизводить) обеспечивает визуальное отображение графической информации на экране электронно-лучевой трубки. Как мы уже упоминали, дисплей может быть векторным или растровым в зависимости от принципа сканирования (произвольного или растрового).

Векторные дисплеи. Первыми появились векторные дисплеи, представлявшие собой усовершенствованный вариант электронно-лучевой трубки (ЭЛТ) с электростатическим отклонением луча. Координаты точки, в которую должен быть выведен луч, задаются с по-

мощью двух чисел, значения которых цифро-аналоговым преобразователем переводятся в напряжения, прикладываемые к отклоняющим пластинам.

Как правило, в графических дисплеях используются ЭЛТ с разрешающей способностью 0,25 мм, т. е. на этом расстоянии две соседние точки различаются как отдельные. Размеры рабочего поля ЭЛТ — 25X25 см. Применение 10-разрядного двоичного кода дает рабочую область, состоящую из 1024X1024 точек.

Основным недостатком обычных ЭЛТ является невозможность длительного сохранения изображения на экране. Если линия высвечивается один раз, она быстро теряет яркость и исчезает, поэтому в дисплеях применяют катодно-лучевые трубки двух типов — запоминающие с видимым изображением и с регенерируемым изображением.

В *запоминающих трубках с длительным послесвечением* (до 1 часа) изображение сохраняется до тех пор, пока оно не будет стерто с помощью электрического импульса.

Регенерировать контурное изображение на экране ЭЛТ можно за счет периодического высвечивания контура (повторения *цикла регенерации* с частотой не менее 30 раз в секунду, при этом развертка, т. е. траектория перемещения развертывающего элемента, имеет случайную форму, она определяется самим изображением).

Наибольшая скорость вычерчивания линий у векторных дисплеев составляет 2 см/мкс. Поэтому число векторов, выводимых на экран без ущерба для качества изображения, ограничено (при частоте, меньшей 10 Гц, изображение мелькает). В лучших системах за один цикл на экран выводится 4—6 тыс. векторов длиной 2—3 см.

Растровые дисплеи. Сталкиваясь с серьезными трудностями при разработке векторных дисплеев, а именно при попытках получить тоновые изображения, разработчики все чаще обращали свои взоры к телевизору — почему бы его не использовать в качестве графического дисплея? Телевизор — это тон, цвет, относительная дешевизна, отработанная технология массового производства.

Так на основе телевизионного устройства и цифровой полупроводниковой техники были созданы растровые дисплеи. (Слово "растр" восходит к латинскому

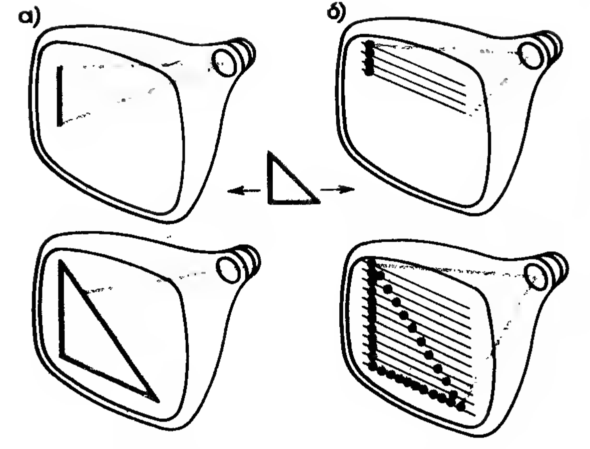


Рис. 8. Построение линии на дисплее:
***а* — векторном; *б* — растровом**

rastrum — грабли, мотыга. Растром обычно называют регулярные, упорядоченные структуры — решетки, чередование прозрачных и непрозрачных полос и т. п., очевидно, по сходству со следом граблей, имеющим вид структур из параллельных борозд).

Экран растрового дисплея разбит на фиксированное число точек (элементов отображения), которые образуют матрицу из фиксированного числа строк и столбцов. Размеры матрицы могут достигать 1024×1280 элементов. Изображение на экране дисплея создается путем избирательной засветки электронным лучом определенных точек экрана. Электронный луч периодически сканирует, обегает экран, как правило, с частотой 25 или 50 раз в секунду. За цикл регенерации формируется изображение, которое может целиком или частично изменяться в реальном времени.

На экране растрового дисплея невозможно начертить непрерывную линию, так как можно высветить только фиксированные точки, которые определяются координатной сеткой. Таким образом, высвечивается последовательность точек, которые лежат приблизительно на заданном отрезке прямой линии (рис. 8). Чем выше разрешающая способность экрана, тем меньше искажена линия.

Какому же устройству отдать предпочтение — векторному или растровому? Векторные дисплеи не тре-

буют больших объемов памяти (а следовательно, они дешевле растровых), позволяют получить четкие контурные изображения, которые можно непрерывно изменять в реальном масштабе времени. Главным их недостатком является невозможность изобразить сплошные области. Двумерные и трехмерные объекты могут быть представлены только в виде "проволочных" каркасов. Более того, если на зкране окажется чересчур много графических элементов и они не будут успевать обновляться за время, отведенное на один цикл регенерации, то частота циклов снизится и изображение начнет мерцать.

Главным достоинством растровых дисплеев является возможность получения тоновых и цветных изображений за счет высвечивания всех точек, лежащих внутри сплошной области. Растровые дисплеи привлекают простотой конструкции, стабильностью изображения (зкран не мерцает вне зависимости от числа графических элементов), но они требуют бóльших объемов памяти, чем векторные.

Графические дисплеи на основе ЭЛТ — самые массовые. В то же время интенсивно развиваются устройства отображения на плоских (в частности — плазменных) панелях. Плазменная панель состоит из двух спаянных стеклянных поверхностей с напыленными или вытравленными в них матрицами злектродов, пространство между которыми заполнено газом. Прецизионный спай обеспечивает взаимную параллельность стеклянных подложек и герметизацию панели. Когда импульс адресного напряжения прикладывается к точке пересечения электродов, газ между электродами ионизируется и плазма начинает светиться. Разряд прекращается путем приложения переменного адресного напряжения, находящегося в противофазе, которое кратковременно понижает уровень рабочего напряжения. Разрешающая способность устройства — 24 лин/см. Максимальный размер матрицы — 512×512 злементов. Такие достоинства, как портативность и виброустойчивость, позволят плоским панелям в будущем успешно конкурировать с ЭЛТ. Но в настоящее время по своим техническим характеристикам они все еще уступают последним.

Графопостроители и принтеры. Различие между этими двумя классами устройств вывода состоит в принципе регистрации информации. Сущность разли-

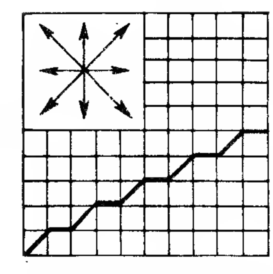


Рис.9. Построение линии на графопостроителе

чия можно понять из названий: графопостроитель *"строит"* т. е. вычерчивает, а принтер (от англ. *print* — печатать) *печатает*.

Графопостороитель. Это устройство служит для вывода на бумагу графической и текстовой информации. Как и дисплеи, графопостроители делятся на две большие группы: с произвольным (векторные) и с растровым (растровые) сканированием. К первой группе относятся планшетные, рассчитанные на бумажные листы, и рулонные графопостроители, в которых в качестве носителя информации применяется широкая бумажная лента. Регистрация производится в прямоугольной системе координат. Пишущим элементом могут служить перо, карандаш, фломастер и даже кисть. Чаще используется стержень шариковой ручки. При этом толщина записываемой линии составляет 0,2—0,6 мм.

В планшетных графопостроителях пишущая головка может перемещаться на плоскости по двум взаимно перпендикулярным направлениям. В графопостроителях рулонного типа перо может перемещаться только в одном направлении, а перемещение в перпендикулярном направлении реализуется за счет движения бумаги. Изображение строится за счет приращений, т. е. движение пишущего устройства (пера) носит характер последовательных малых (0,025—0,25 мм) перемещений — шагов, поэтому изображение состоит из множества мелких штрихов. Обычно движение пера ограничено восемью направлениями (рис. 9).

На рис. 10 показан *многоцветный* векторный графопостроитель. В нем пишущие элементы располагаются вдоль края плоскости черчения. Цвет линии задают с помощью специальной программы. Для перехода с одного цвета на другой пишущее устройство подходит к краю плоскости и устанавливает на место пишущий элемент. Затем захватывает пишущий элемент другого цвета и возвращается на нужную позицию.

Например, с помощью планшетного графопостроителя ЕС-7051 можно получать трехцветное изображение. Пишущими элементами служат специальное перо, наполненное чертежными чернилами, либо фломастер,

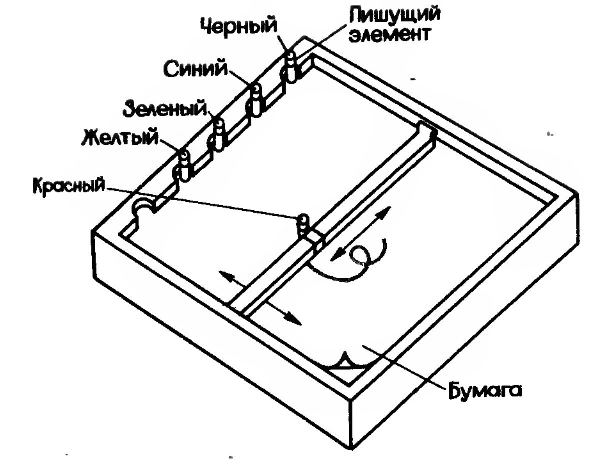


Рис. 10. Планшетиый миогоцветный графопостроитель

шариковая ручка. Скорость вычерчивания — 50 мм/с, шаг — 0,05 мм, размеры рабочего поля — 1050×1000 мм. Объем памяти оперативного запоминающего устройства – 4 кбайта.

Графопостроители обеспечивают построение изображений более высокого качества (более точных) по сравнению с графическими дисплеями, но они имеют гораздо меньшее быстродействие, что связано с инерционностью движения механических узлов.

Чтобы повысить скорость регистрации, нужно отказаться от последовательного вычерчивания каждой линии и перейти к их точечному представлению, т. е. к растровому сканированию.

Примером растрового графопостроителя является *струйно-чернильный*. Печатающая головка состоит из четырех форсунок, наполненных чернилами красного, синего зеленого и черного цветов. Черный цвет используется потому, что при печати с помощью комбинации трех основных цветов трудно добиться насыщенного черного цвета. При подаче напряжений на пьезокристалл происходит "выстрел": в камере с чернилами возникает волна давления, которая проталкивает чернила через микроскопическое отверстие форсунки. На выходе отверстия формируется капля чернил, которая летит к бумаге, натянутой на быстровращающийся барабан, со скоростью несколько метров в секунду. Печатаю-

щая головка сканирует бумагу, время сканирования лежит в пределах от 45 с до 4,5 мин. При частоте печати 5 кГц может быть обеспечена разрешающая способность 9,5 лин/мм. Пример цветного изображения, полученного на струйно-чернильном графопостроителе, приведен на рис. 11.

Принтер. Печатающая головка растрового принтера содержит 7 или 9 близко расположенных по вертикали штырьков. За один проход головка может напечатать строку, содержащую порядка 320 точек по длине (в принтерах персональных компьютеров) и соответственно 7 (9) точек по ширине строки. Таким путем независимо от длины и количества выводимых графических элементов обеспечивается постоянная скорость вывода графической информации.

Появляются сообщения о выпуске дешевых цветных точечно-матричных печатающих устройств. В них, как в пишущих машинках, используется многоцветная лента, но здесь она разделена не на две, а на четыре цветные полосы: красную, синюю, желтую и черную. В процессе работы лента устанавливается так, чтобы требуемая цветная полоса располагалась между печатающей головкой устройства и бумагой.



Рис. 11. Изображение, полученное на струйно-чернильном графопостроителе

Если в векторном графопостроителе наименьшим выводимым элементом является линия, то в растровом графопостроителе, так же как и в растровом дисплее, — точка. Это обстоятельство натолкнуло конструкторов на мысль о создании универсального устройства, соединяющего графопостроитель и принтер, и позволяющего выводить как текстовую, так и графическую информа-

цию. Большие возможности растровой регистрации изображений и текстов открывает развитие *лазерных* принтеров. Уже существуют ксерографические лазерные принтеры которые позволяют получать высококачественные копии с разрешением 14 000 точек/см^2. Скорость регистрации достигает 12 страниц в минуту. В лазерных принтерах осуществляется высокоскоростная развертка лазерным лучом.

Как хранится изображение? Многие художники обладают удивительной зрительной памятью. Так, известный художник Ге изобразил по памяти комнату петровского Монплезира на картине "Петр I допрашивает царевича Алексея Петровича в Петергофе", которую он видел всего лишь один раз. По памяти писал свои картины Айвазовский, объясняя это тем, что изобразить с натуры молнию, порыв ветра, всплески волн немыслимо. Художник должен запомнить их.

Объяснить этот феномен наука пока не может, как и многие другие загадки, связанные с человеческой памятью. Зато мы можем приоткрыть читателю секреты машинной памяти, которая по емкости приближается к человеческой, а по надежности хранения и точности воспроизведения превосходит ее. Мы не будем рассматривать оперативное запоминающее устройство компьютера, поскольку оно универсально, т. е. позволяет хранить любую информацию, представленную в числовом виде — графическую, текстовую, звуковую. Нас интересует только графическая информация. Тем же, кто захочет поподробнее ознакомиться с устройством и работой машинной памяти, мы рекомендуем книгу В.В. Александрова, В.Н. Арсентьева, А.В. Арсентьевой "Что может ЭВМ?" ("Машиностроение", 1986).

Способ хранения графической информации определяется типом используемого в графической системе дисплея. Если используется векторный дисплей, то каждой точке рисунка должны соответствовать два числа — координаты x и y. Таким образом, ввести рисунок, предназначенный для воспроизведения на векторном дисплее, — это значит записать в память ЭВМ некоторую последовательность (массив) чисел — значений x и y. При воспроизведении рисунка выполняется обратная процедура: последовательно просматриваются два массива значений x и y, и пары чисел x и y подаются на устройства вывода. При этом луч вы-

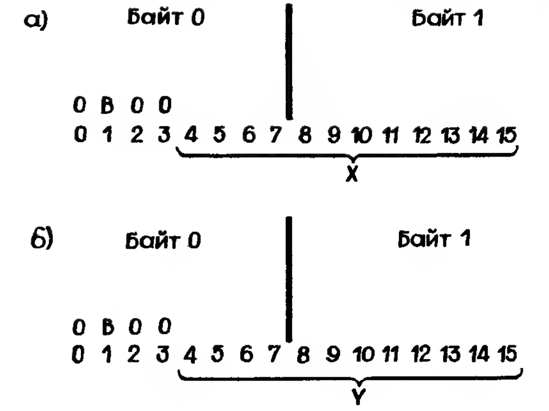


Рис. 12. Представление координат точки в памяти компьютера для векторного дисплея:
а **— координаты** ***X*** **;** ***б*** **— координаты** ***У*** **(при В=1 луч выключен; при В=0 включен)**

свечивает на экране ЭЛТ точку, соответствующую определенной паре чисел.

Предположим, что на рабочем поле экрана выделено 1024Х1024 точек, в которые может быть направлен электронный луч. Тогда каждая координата представляется 12-разрядным двоичным числом и занимает в памяти полуслово (рис. 12). Первый бит полуслова с координатой *x* служит для управления яркостью луча: если в этом бите записана единица, луч переводится в новую точку в выключенном состоянии.

Чтобы воспроизводить изображение на растровом дисплее, приходится хранить информацию не о координатах точек изображения, а о значениях интенсивности высвечивания всех элементов отображаемой картинки.

Способ хранения и записи изображения, при котором на кодирование значения яркости или цвета каждого элемента отводится один или более бит называется созданием *битовой карты изображения* (рис. 13). Битовая карта изображения записывается в запоминающее устройство системы отображения, которое называется *видеопамятью*.

Возможны различные варианты организации памяти. Возьмем для примера память объемом 1 Мбит (131 кбайт). При кодировании элемента изображения

одним битом ее можно организовать в виде плоскости с размерами 1024×1024×1 бит. Это позволит нам получать только контрастные (черно-белые) изображения. При организации памяти в виде четырех плоскостей с размерами 512×512×4 бит на один элемент изображения приходится 4 бита. Для получения реалистического черно-белого изображения обычно требуется 32—64 градации тона, т.е. одному элементу должно соответствовать 5—6, а может и 8 бит, а для цветного изображения примерно в три раза больше (по 8 бит на каждый) из трех основных цветов — красный, синий, зеленый, т.е. всего 24 бит.

Несмотря на значительное снижение стоимости памяти, системы, в которых на 1 элемент изображения приходится 24 бита, все же довольно дороги. Поэтому создатели графических систем, особенно массового пользования на основе персональных компьютеров, стоят перед альтернативой: либо высокая разрешающая способность по плоскости, либо богатство "палитры" отображаемых цветов или полутонов.

Особенностью систем хранения графической информации являются большие объемы машинной памяти.

В настоящее время в качестве носителей информации широко применяются *магнитные ленты* и *диски*. Информация записывается и считывается с помощью

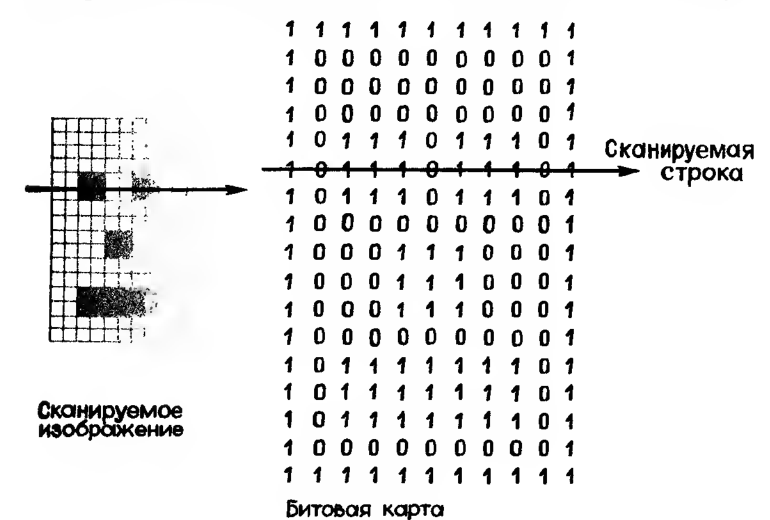


Рис. 13. Битовая карта изображения в запоминающем устройстве компьютера

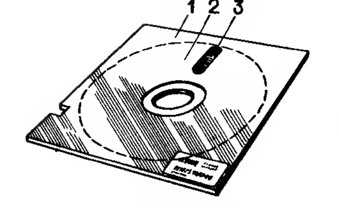


Рис. 14. Гибкий магнитный диск:
1 — пластиковый конверт; 2 — магнитный диск; 3 — прорезь для головки записи — считывания

магнитной головки приблизительно так же, как в магнитофоне. Только в данном случае записываются последовательности фиксированных импульсов, соответствующих логическим нулям и единицам. В современных устройствах поперечная плотность записи достигает 8 дорожек на 1 мм, продольная — 160 бит/мм.

Для персональных компьютеров основным видом устройств внешней памяти являются накопители на гибких магнитных дисках. Это дешевые, быстродействующие, малогабаритные устройства. Носителем информации является круглая гибкая пластмассовая пластина диаметром 19,8 см, на которую нанесен магнитный слой, она находится в защитном чехле — пластиковом конверте (рис. 14).

Информация записывается либо на одной стороне диска, либо на двух. На одном диске можно хранить от 125 кбайт до 500 кбайт информации. Но и это не предел.

Интенсивно развиваются накопители на *оптических дисках*, у которых считывание и запись данных осуществляются с помощью луча лазера. Оптический диск имеет двухслойную конструкцию: металлический носитель записи заключен в пластмассовую оболочку. Чтобы записать информацию на диск, металлический слой нагревают лучом лазера. При этом происходит локальное разложение полимера, в металлическом слое образуется микроскопическая выпуклость. Такие выпуклости при воспроизведении обнаруживаются точечным лучом лазера.

По плотности записи оптические запоминающие устройства превосходят традиционные устройства на магнитных носителях в 100 раз. Информационная емкость оптических дисков диаметром 50—405 мм достигает 4 Гбайт. Существуют ленточные системы оптической памяти емкостью 600 Гбайт. Такой объем данных, размещаемых в устройстве, которое по размерам несколько больше, чем тумба письменного стола, эквивалентен 250 млн стандартных машинописных страниц.

Лента длиной 250 м, шириной 4 мм и толщиной 12 мкм размещается на небольших кассетах. Устройство содержит 100 таких кассет емкостью 6 Гбайт каждая, смонтированных на карусели. В центре карусели неподвижно установлены 16 устройств для записи и воспроизведения информации при помощи лазерных диодов. Карусель вращается, подавая требуемую кассету к одному из устройств записи—воспроизведения. В настоящее время существует техническая возможность для переноса на оптические диски Большой Советской Энциклопедии. Такие диски скоро можно будет использовать при работе с персональным компьютером.

2. ЭЛЕКТРОННЫЙ ЧЕРТЕЖНИК

Чем работа любой машины отличается от труда человека? Прежде всего — высокой повторяемостью результатов. Особенно ясно это видно на примере компьютера. Если компьютер рисует, то все рисунки будут абсолютно идентичны. Первый рисунок вы не отличите от тысячного. Естественно поэтому, что в первую очередь графические системы стали применяться для черчения, хранения введенных в память машины чертежей, изготовления копий чертежа. Оказалось, кроме того, что графические системы очень удобны для выполнения надписей, вычерчивания стандартных узлов, решения задач на построение (прямых, взаимно параллельных или перпендикулярных линий, концентрических или соприкасающихся окружностей, сопряжений, скруглений).

При оформлении чертежей эти работы занимают значительную часть времени и требуют специальных навыков. Многие инженеры и конструкторы испытывают неприязнь к такого рода работе, считая ее рутинной, нетворческой. И здесь на помощь может прийти компьютер.

Чертежи, получаемые с графопостроителя, могут быть непосредственно использованы на производстве для изготовления деталей, печатных плат, раскроя материалов и т.п. В этом случае особо высокие требования предъявляются к точности регистрации графической информации. Размеры выводимых чертежей могут достигать 7,0Х 7,5 м.

Так что же должен "знать" и "уметь" компьютер, чтобы выполнять функции чертежника?

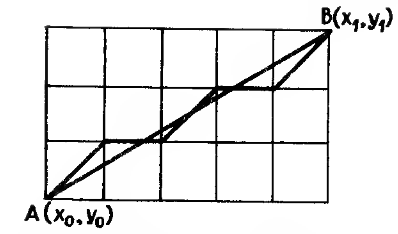


Рис. 15. Аппроксимация прямой линии в векторной графике

Первая линия — прямая. Программа построения прямой линии является тем основанием, на котором строится все графическое программное обеспечение[1]. Если компьютер "умеет" чертить отрезки прямых, то его можно научить выполнять почти любую графическую работу.

Как уже говорилось выше, в векторном дисплее или графопостроителе из каждой данной точки можно выполнить переход в одну из 8 фиксированных соседних точек (рис. 15). Поэтому при переводе пера из точки *A* с координатами x_0, y_0 в точку *B* с координатами x_1, y_1 необходимо задать такую последовательность движений, которая бы наилучшим образом приближалась к прямой.

Аналогичная проблема возникает и в растровой машинной графике. Построение прямой на экране дисплея осуществляется в принципе примерно так же, как на бумаге, т.е. сначала отмечают начальную и конечную точки отрезка, а затем их соединяют прямой линией.

Команды вызова программ построения отрезков различны в разных графических машинных языках. Мы не можем привести здесь все эти команды, да в этом нет и необходимости. Гораздо важнее понять основные принципы построения отрезков, это поможет вам ориентироваться в новом для вас языке.

Предположим, что вы работаете с векторным графопостроителем и пользуетесь комплексом графических программ ГРАФОР, написанных на языке ФОРТРАН. (Этот комплекс широко применяется для больших и средних ЭВМП.) Перед описанием графического изображения вы должны определить границы, формат "страницы", т.е. прямоугольного поля, в пределах которого на листе бумаги должно размещаться описываемое графическое изображение. Перо графопостроителя может быть выведено за пределы страницы только после того, как страница закрыта.

---

[1] Программное обеспечение — это совокупность машинных программ. Обычно поставляется фирмой-разработчиком вместе с компьютером.

Примем формат страницы равным 10X15 см. Теперь для того, чтобы начертить отрезок прямой или просто переместить перо в заданную точку чертежа, нужно обратиться к графической программе нижнего уровня — подпрограмме MOVE (X, Y, Z). Координаты (x_0, y_0) считаются координатами текущего положения пера.

Подпрограмма MOVE позволяет перевести перо в другую точку с координатами (x, y). Левый нижний угол страницы принимают в качестве точки отсчета, его координаты — нулевыми, координаты всех других точек задаются относительно этого угла в единицах, которые были выбраны перед определением страницы. Параметр Z определяет состояние пишущего элемента: при Z = 1 пишущий элемент опущен во время движения, при Z = 0 — поднят.

Из уроков черчения вы знаете, что на чертежах применяются линии различного начертания: сплошная (основная и тонкая), штриховая, штрих-пунктирная. На графопостроителе тип линии задают с помощью программы.

Увеличение толщины линии достигается либо за счет многократного вычерчивания, либо за счет переключения пишущего узла.

Возьмем другой вариант. Вы — пользователь персонального компьютера. Например, большинство рисунков к данной книге выполнено с помощью персонального компьютера PC IBM, а программы написаны на языке БЕЙСИК. В данном случае рабочей областью построения изображений является весь экран дисплея. Значения отсчетов по осям определяются разрешающей способностью экрана. Значения x определяем по горизонтальной оси от левого края экрана, выбирая одно из 320 возможных положений, значение y отсчитываем от верхнего края экрана, оно может изменяться от 0 до 199. Для вызова программы построения отрезков применяется команда LINE(X1, Y1)—(X2, Y2), C, где X1, Y1 — координаты начальной точки отрезка; X2, Y2 — конечной; C — параметр, задающий цвет линии (при C=1 вычерчивается линия синего цвета, при C=2 — зеленого и т.д.; при отсутствии C строится черно-белая линия).

На рис. 16 показаны три отрезка, построенные таким образом из одной исходной точки. А вот как выглядит программа их построения:

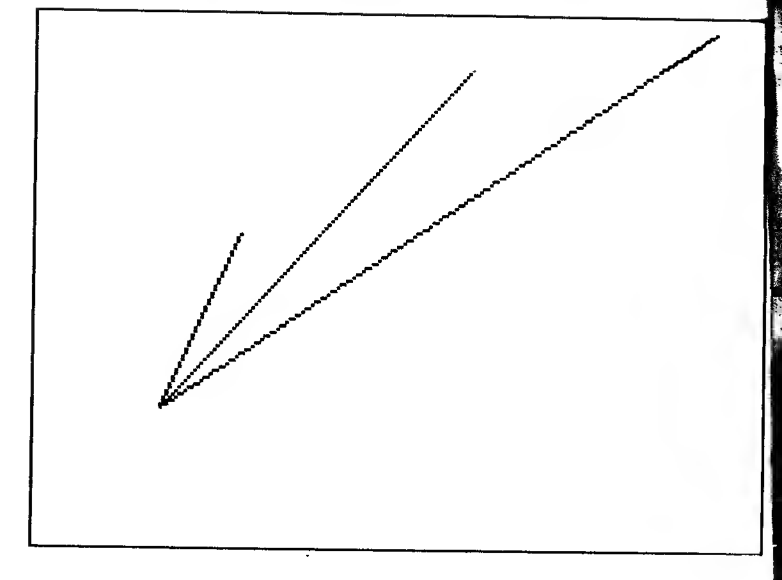


Рис. 16. Семейство из трех отрезков

```
10 LINE(50,150)-(75,100)
20 LINE(50,150)-(150,50)
30 LINE(50,150)-(250,25)
40 LINE(10,10)-(300,190),,B
```

Рис. 17 дает представление о том, как выглядит на экране дисплея прямая при различных угловых положениях.

Электронный циркуль. В программном обеспечении практически любой графической системы имеется программа построения окружности. Для ее построения достаточно задать три параметра — координаты центра окружности и радиус. Обращение к подпрограмме — CIRCLE(X, Y), R, где X, Y — координаты центра окружности; R — радиус.

Окружности, изображенные на рис. 18, построены с помощью следующей программы:

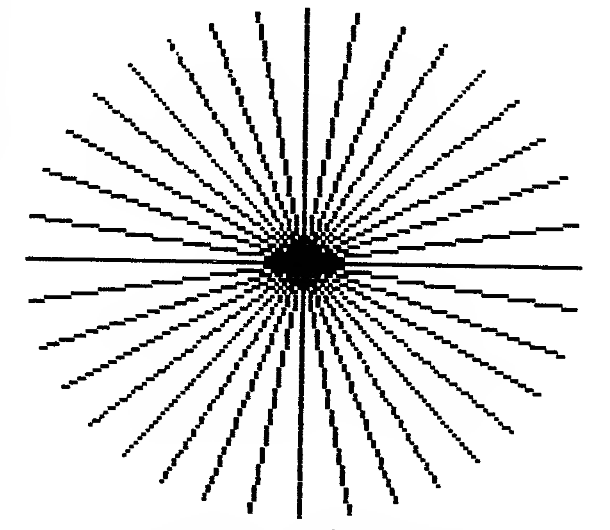


Рис. 17. Семейство отрезков, проходящих через точку под разными углами

```
20 FOR R=10 TO 60 STEP 20
30 CIRCLE(160,100),R
40 NEXT
```

Строка 20 задает начало цикла, строка 30 содержит вызов подпрограммы CIRCLE, строка 40 — конец цикла. В операторе цикла переменная R (радиус окружности) получает значения 10, 30 и 50.

Выбор цвета осуществляется так же, как в подпрограмме LINE,с помощью параметра C: CIRCLE(X, Y), R, C. Но если такой программы нет, то ее несложно и написать.

Как изобразить окружность с помощью линейки? Нужно построить многоугольник с очень большим числом сторон. Такое построение легко выполнить на компьютере. Вычислим координаты 100—500 точек окружности с помощью тригонометрических функций синуса и косинуса, соединим полученные точки отрезками прямых. Для этого используем графические программы, подобные упомянутым выше MOVE и LINE.

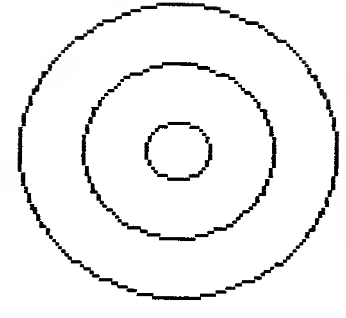


Рис. 18. Семейство окружностей

Электронное лекало. Приходилось ли вам работать с лекалом? Соединить точки плавной кривой не так просто. Поскольку заранее не известно, какое лекало нужно, приходится работать по методу проб и ошибок. Чтобы применить компьютер для решения этой задачи, ее необходимо формализовать.

В математике задача приближенного описания функций называется *аппроксимацией*. Для построения функций (т.е. кривых линий) необходимо задавать координаты точек. Имея программу построения прямых, можно строить любые кривые, используя аппроксимацию полигональными линиями. Однако чтобы аппроксимирующая кривая была максимально близка к описываемой функции, требуется задать координаты как можно большего числа точек. В противном случае велика ошибка аппроксимации (что особенно важно при черчении), к тому же изображение незстетично. Чтобы получить плавную кривую, при минимальном числе точек, можно воспользоваться злектронными лекалами — *сплайнами*. Что это такое?

Математические сплайн-функции берут свое начало от тонких и гибких стержней, которыми пользовались чертежники для проведения плавных кривых через заданные точки. Стержень закреплялся в нескольких точках и изгибался таким образом, что обеспечивалась минимальная "знергия" натяжения (рис. 19).

Сравните два изображения на рис. 20. Левое получено методом полигональной аппроксимации (контур головы построен по 20, уха — 9, очки — 12 точкам). Правое изображение построено по тем же опорным точкам с помощью сплайнов, оно зстетичнее.

Штриховка. Контурное изображение становится нагляднее, если его закрасить, заштриховать. Выполнение штриховки на чертеже или тонирование (растушевка) плаката — трудоемкая и утомительная работа. Требуются большая аккуратность и адское терпение: при штриховке расстояния между линиями должны выдерживаться строго постоянными, нельзя заезжать за гра-

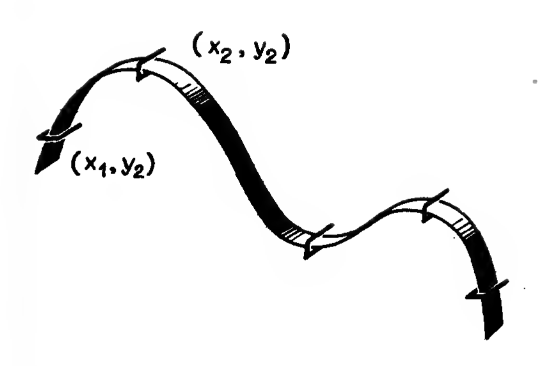


Рис. 19. Физический аналог сплайн-функций

ницы штрихуемой области; при растушевке требуется обеспечить равномерность тона (за счет постоянства усилия прижима).

Стремясь облегчить труд конструктора или дизайнера, разработчики программного обеспечения графических систем предложили передать эти функции компьютеру. При работе в интерактивном режиме области можно закрашивать непосредственно на экране. Для этого пользователь предварительно очерчивает закрашиваемую область, а затем указывает световым пером или курсором на любую ее точку, выбрав предварительно цвет закраски. Существуют графические системы, в которых пользователь перемещает световое

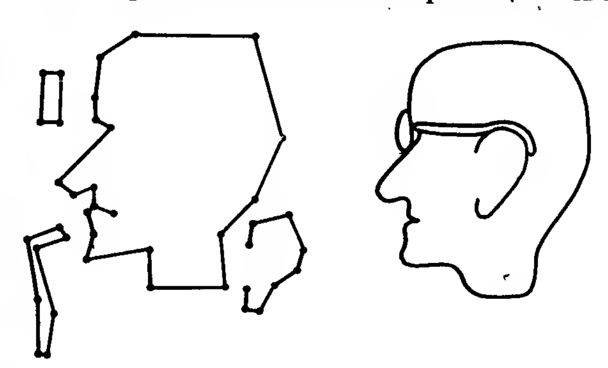


Рис. 20. Аппроксимации:
линейная (слева) и кубическими сплайнами (справа)

перо или курсор, закрашивая изображение, как кистью. Цвет задают с помощью функциональной клавиши. Чтобы работа шла быстрее, можно увеличить курсор, взять "кисть" потолще (рис. 21).

Границы закрашиваемой области могут быть вычислены и с помощью специальной программы. Подпрограмма построения прямоугольника (рис. 22) в языке БЕЙСИК имеет такое же имя, как и подпрограмма построения отрезка линии, — LINE, но с добавлением параметра B (BOX — ящик):

```
10 LINE(50,150)-(70,170),,B
20 LINE(100,150)-(120,170),,BF
```

Отрезок, задаваемый параметрами X1, Y1 и X2, Y2, представляет собой диагональ прямоугольника и не вычерчивается. Параметр F в строке 20 показывает, что прямоугольник должен быть закрашен (FILL — наполнение). Управление цветом закраски осуществляется так же, как в подпрограмме построения линии, т.е. за счет введения параметра цвета C.



Рис. 21. Электронная кисть

Выполнение надписей. Выполнение надписей на чертежах и документах — очень трудоемкий процесс. И здесь освободить время и знергию конструктора для творческих задач помогает компьютер.

И в векторных, и в растровых устройствах знаки формируются на основе сетки (рис. 23). Только в векторном представлении литера составляется из отрезков прямой, а в растровом — из точек. Размер сетки должен быть не менее 5X7 элементов. Такая сетка содержит 35 точек, поэтому любую литеру можно представить 35 битами. Среднее число точек, требуемое для изоб-



ис. 22. **Закраска прямоугольника на принтере (*а*) и штриховка на графопостроителе (*б*)**

ажения литер печатного текста, равно 20, прописных строчных — 45.

Каждому символу соответствует своя начальная очка, которая используется для привязки при записи имвола в запоминающее устройство.

В ГРАФОРЕ программу построения знаков вызы-ают с помощью команды SYMBOL. Она позволяет вы-ести любую последовательность знаков, любого раз-ера, на любое место страницы. Общее число различ-ых знаков составляет 176. Все они разбиты на четыре бора — прописные латинские и русские буквы, строч-ые латинские и русские буквы, прописные греческие, трочные греческие буквы. Тексты могут быть напи-аны как прямым шрифтом, так и курсивом с правым ли левым наклоном. Таким образом, вы можете на-рать на клавиатуре школьное сочинение или решение атематической задачи, а компьютер его красиво на-ишет — все буквы будут одинакового размера, состав-ены в ровные строчки, а в формулах будут аккуратно роставлены все символы и знаки.



Рис. 23. Формирование знаков:
а — **на графопостроителе;** *б* — **на принтере**

example
example
example
example
example
example
example

example
example
example
example
пример

Рис. 24. Образцы шрифтов, печатаемых на принтере

О возможностях растровой печати можно судить по рис. 24. (Мы пользовались принтером с высокой разрешающей способностью.) Подобные шрифты могут использоваться не только для оформления чертежей, но и таблиц, диаграмм, графиков.

3. КОМПЬЮТЕР С ПРОСТРАНСТВЕННЫМ ВООБРАЖЕНИЕМ

Рисуют головой, а не руками.
Микеланджело

В IV веке до н.з. в Древней Греции существовала Сикионская школа рисунка. Там учеников призывали изучать природу и ее закономерности на основе математики. На дверях Сикионской школы рисунка было написано: "Сюда не допускаются люди, не знающие геометрии". И это не случайно: знание геометрии особенно необходимо для отображения трехмерного мира на плоскости.

А теперь прочтите следующую фразу: "Если зрение не может достигнуть предметов по прямым линиям, то этих предметов нельзя увидеть, так как зрение не воспринимает ничего по кривым линиям". Заключенная в ней истина настолько очевидна, что декларировать ее кажется смешным. А ведь она взята из учебника, написанного великим немецким художником А. Дюрером. Здесь есть над чем призадуматься. Мы так свыклись с законами перспективы и даже не можем себе представить, что люди когда-то их не знали!

Теперь когда мы "научили" компьютер чертить, можно "развивать его пространственное воображение".

Трехмерный мир в двумерном исполнении. Долгий путь, измеряемый многими тысячелетиями, прошло развитие рисунка и живописи, прежде чем были открыты основные законы реалистического отображения окружающей нас действительности — законы перспективы и светотени. До открытия этих законов живопись носила плоскостной характер.

Так, живописцы Древнего Египта на протяжении тысячелетий пользовались почти одними и теми же приемами. Пространство, отделяющее дальние фигуры от стоящих ближе к зрителю, египетские художники старались передать условным расчленением на горизонтальные пояса (зоны): нижний пояс относился к ближнему плану, самый верхний — к дальнему.

Первые правила перспективы были сформулированы древнегреческим математиком Эвклидом (III в. до н.э.). Дальнейшее их развитие было связано с практическими достижениями итальянской живописи эпохи Возрождения. Леонардо да Винчи, в совершенстве овладевший искусством перспективы и светотени, писал в своем "Трактате о живописи": "Перспектива есть не что иное, как вид из окна, на совершенно прозрачном стекле которого изображены предметы, находящиеся за окном".

Заметим, что слово "перспектива" употребляется в двух значениях. Во-первых, это наука о закономерностях изображения на плоскости трехмерного мира в соответствии с особенностями глаза как оптической системы и физиологии зрения человека. Во-вторых, перспективой называют и само изображение, выполненное в соответствии с положениями этой науки.

Рассмотрим пример центральной перспективы. Если сесть на расстоянии вытянутой руки от стекла, то можно, закрыв один глаз и не меняя положения головы, пером или кисточкой "обвести" на его плоскости то, что мы видим за ним, и таким образом получить рисунок, картину видимого. Каждый раз, когда кисть рисует на стекле какую-нибудь видимую точку, она тем самым отмечает место прохождения луча, направленного в глаз от этой точки. Вот такая плоскость, на которой изображаются видимые предметы в том порядке, как они воспринимаются наблюдателем, называется *картинной* (рис. 25). Точки на горизонте, в которых

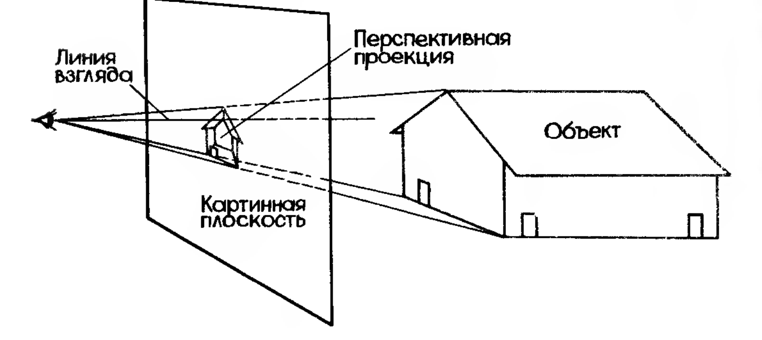


Рис. 25. Центральная перспектива

зрительно сходятся параллельные линии, уходящие в глубь картины, называются *точками схода. Линия взгляда* (зрения) проходит перпендикулярно к картинной плоскости. Перспективной проекцией является пересечение линий взгляда с картинной плоскостью. Размер перспективной проекции определяется расстоянием картинной плоскости от глаза наблюдателя: чем дальше картинная плоскость от глаза наблюдателя, тем больше размер проекции.

На рис. 26 приведен вариант перспективной проекции. Это так называемая *"кривая"* проекция. Линии, параллельные в трехмерном пространстве, пересекаются на проекции. Так, линии, параллельные оси x, сходятся в точке x_0, параллельные оси y — в точке y_0. Линии, параллельные оси z, не пересекаются.

При отображении технических объектов широко применяются *прямоугольные* проекции. Прямоуголь-

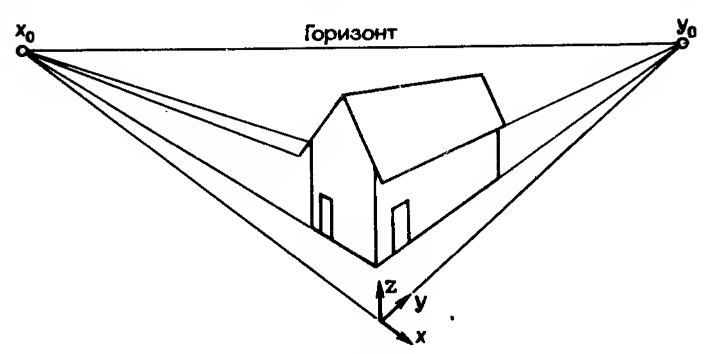


Рис. 26. Вариант перспективной проекции

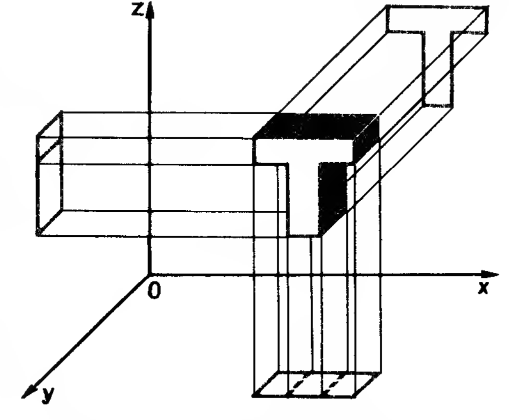


Рис. 27. Прямоугольное проецирование

ные прекции — чертежи (виды сверху, спереди и сбоку), формируемые в результате проецирования в направлениях, параллельных координатным осям. Прямоугольную проекцию можно рассматривать как центральную перспективу с бесконечно удаленным центром проецирования. Построение прямоугольной проекции, например в направлении оси z на плоскость проекций, перпендикулярную к названной оси и находящуюся на расстоянии z_0 от начала координат, сводится к присвоению значения z_0 всем координатам z (рис.27).

Точка зрения. Построение перспективы начинается с выбора *точки зрения*, т.е. того места, откуда рисующий смотрит на изображаемый предмет. Если на одну и ту же сцену смотреть, то выпрямляясь в полный рост, то присаживаясь, то передвигаясь налево или направо, она будет представляться каждый раз по-иному.

Помните:

Вороне где-то бог послал кусочек сыру.
На ель ворона взгромоздясь,
Позавтракать уж было собралась,
Да призадумалась ...

О чем призадумалась Ворона, нам неизвестно, а вот художнику В. Серову, когда он приступил к иллюстрации этой басни, было над чем поломать голову. Чтобы узнать, как выглядела местность с точки зрения наблюдателя (Вороны), Серов приставил лестницу к громадной ели и, забравшись на один из верхних суков, на котором по его представлению, должна была сидеть

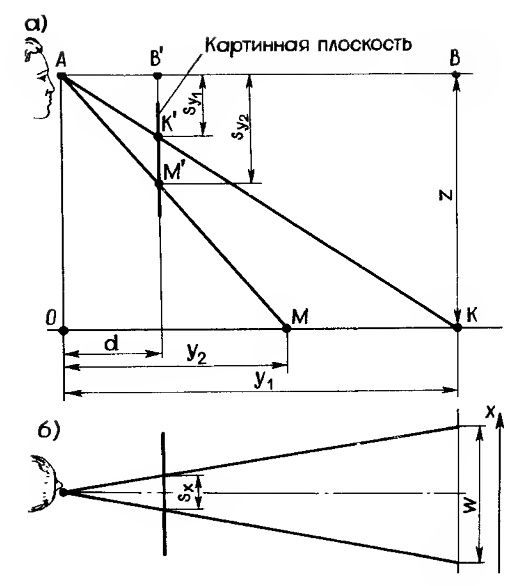


Рис. 28. Перспективные вычисления

Ворона, рискуя упасть, делал оттуда зарисовки. Если бы в то время существовали графические системы, то В. Серову было бы достаточно обмерить полянку и ввести эти данные в компьютер. После этого, задав точку наблюдения, он мог бы обозревать полянку с той же точки зрения, что и Ворона, но без риска упасть с дерева.

Мы привели шуточный пример. Но далеко не шуточной эта проблема может стать, например, при проектировании аэродромов или подготовке летчиков. Как будет выглядеть взлетно-посадочная полоса с разной высоты и под различными углами зрения? А вот с помощью компьютера мы может окинуть "мысленным взором" аэродром из любой возможной точки зрения.

Формулы перспективы. Чтобы "научить" компьютер перспективным построениям, их нужно формализовать.

Представьте себе, что вы стоите на полотне ровной дороги, которая уходит за горизонт. Хотя края дороги в действительности параллельны, нам кажется, что дорога по мере приближения к горизонту становится уже. Уменьшаются автомобили и люди, если они движутся от нас к горизонту. Как же описать эту закономерность математически?

Для начала определим систему осей координат, относительно которых мы будем определять положение точек объектов, наблюдаемых нами, и соответствующих им точек (проекций) на экране дисплея. Положим, что ось x идет слева направо, ось z — вертикальная, а ось y идет вдоль направления взгляда, т.е. перпендикулярно плоскости экрана.

Мы не можем видеть дорогу, если наш глаз находится на уровне дороги, поскольку каждая точка дороги будет лежать вдоль линии, проходящей через ось x. Для того чтобы дорога была видна, точка зрения должна находиться выше ее уровня (точка A на рис. 28, *а* расположена на высоте z от поверхности дороги).

Предположим, что мы смотрим на дорогу через окно (картинную плоскость), расположенное на расстоянии D от нас. Горизонтом является линия, проходящая через наш глаз параллельно дороге, т.е. оси y.

Выберем точку K на дороге на расстоянии y_1. На картинной плоскости она появится ниже горизонта (точка K'). Поскольку треугольник AKB подобен треугольнику $AK'B'$, то

$$S_{y_1}/D = z/y_1 \quad \text{или} \quad S_{y_1} = zD/y_1,$$

т.е. высота изображения на картинной плоскости обратно пропорциональна расстоянию от объекта до наблюдателя y_1. Если мы возьмем на дороге другую точку M, расположенную от глаза на расстоянии y_2 на картинной плоскости ей будет соответствовать точка M', для которой $S_{y_2} = zD/y_2$.

Предположим, что вдоль осевой линии дороги проведена последовательность отрезков равной длины (линия разметки). Каждый отрезок отобразится на картинной плоскости в короткую вертикальную линию. Чем дальше отрезок линии на дороге находится от глаза наблюдателя, тем короче его проекция на картинной плоскости.

Внимательно рассмотрев рис. 28, *б*, вы можете убедиться, что отрезок S_x, отображающий ширину дороги W на картинной плоскости, обратно пропорционален y: $S_x = WD/y$.

Аналогичным образом отображаются на картинную плоскость и вертикальные объекты, расположенные вдоль дороги.

Говорят, что если хочешь что-нибудь понять, надо объяснить это другому. Но "другой" может оказаться

понятливей, схватить все на лету, поправить. Можно предложить другой рецепт: хотите что-нибудь понять — "объясните" компьютеру. Ведь компьютер понимает вас только буквально и в его работе сразу выявятся ваши ошибки.

Представим, что экран дисплея находится в картинной плоскости. Начертим горизонтальную линию через экран, она будет опорной — линией горизонта. Если мы теперь проведем два отрезка из нижних углов экрана к точке, расположенной посередине горизонтальной линии, мы получим проекцию дороги.

Предположим, что дорога имеет ширину $x = 7{,}5$ м и мы наблюдаем ее на экране дисплея шириной $S_x = 30$ см с расстояния $D = 1$ м. Поскольку $S_x = Dx/y$, $y = 25$ м, что соответствует точке начала обзора дороги.

Однако на экране мы откладываем не сантиметры, а отсчеты. Поэтому нам необходимо определить масштабные коэффициенты для нашего экрана по осям x и y.

Масштабный коэффициент S_{cx} для нашего экрана должен быть выбран таким, чтобы при $y = 25$ м, $x = 7{,}5$ м и $D = 1$ м значение S_x на экране дисплея равнялось максимальному количеству отсчетов по оси x, например в нашем случае — 630. Поскольку

$$S_x = S_{cx}Dx/y,$$

а ширина экрана дисплея в нашем случае равна 30 см (0,3 м), то масштабный коэффициент $S_{cx} = 2100$.

Подобным образом можно вычислить масштабный коэффициент по оси y из выражения

$$S_y = S_{cy}Dz/y.$$

Предположим, что точка наблюдения располагается на высоте 3 м, а точка, лежащая на расстоянии $y = 25$ м (в начале обзора), соответствует самой нижней части экрана. Линия горизонта отображается на экране горизонтальной линией, расположенной на высоте 100 отсчетов, т.е. $S_y = 100$. Тогда $S_{cy} = 830$.

Чтобы получить перспективный вид дороги с деревьями на обочинах (рис. 29), необходимо заполнить таблицу значений высот и расстояний до деревьев. Затем координаты их оснований и вершин будут вычислены по приведенным выше формулам.

Изображение поверхностей. Реальные объекты являются, как правило, непрозрачными. Они отражают

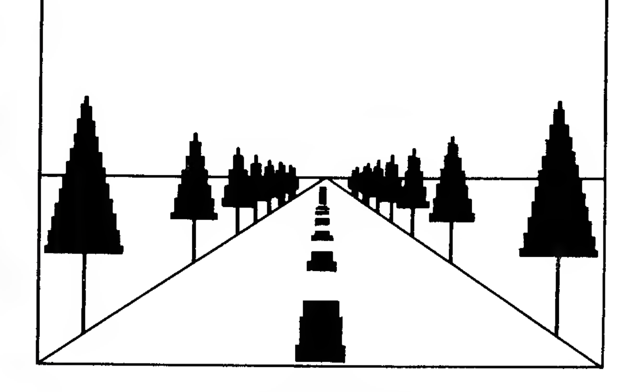


Рис. 29. Перспективное изображение, построен - ное компьютером

свет, бликуют, отбрасывают тени. Это необходимо учитывать в графическом изображении объекта.

Чтобы создать иллюзию непрозрачности изображаемой поверхности, необходимо определить те части объекта, которые скрыты от наблюдателя непрозрачной поверхностью, и не рисовать их на экране.

Проблему определения видимости линий (или проблему ”невидимых линий”) приходится решать не только для контурных изображений, но и для тоновых (рис. 30).

Существуют два аспекта проблемы. Первый заключается в определении невидимых поверхностей и ребер, заслоненных элементами самого тела. Второй аспект — выявление перекрытия частей тела другими телами, находящимися в поле зрения наблюдателя. Мы затронем только первый аспект.

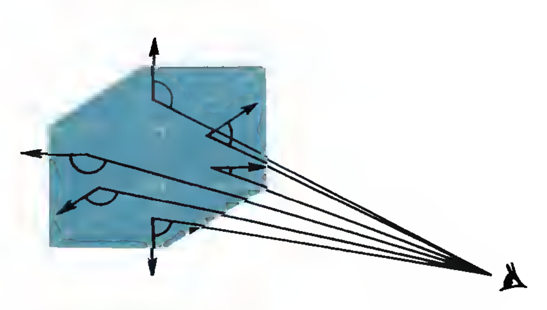


Рис. 30. Определение вндимостн граней

Существуют две группы методов определения видимости элементов объекта: проверка по поверхностям (граням) и проверка по точкам.

Проверка по поверхностям позволяет исследовать видимость поверхности в целом. Поверхность видима, если угол между лучом зрения, выходящим из глаза наблюдателя, и нормалью (перпендикуляром к плоскости, проходящей через точку касания) к соответствующей поверхности полиэдра (многогранника) больше 90°. Ребро, представляющее собой линию пересечения двух невидимых поверхностей, невидимо, и его следует исключить из числа ребер строящегося графического изображения. Процедура проверки по поверхностям весьма эффективна, поскольку после определения видимости поверхностей необходима только одна проверка для каждого ребра, но она применима только к ограниченному числу объектов — выпуклым полиэдрам.

Широкий класс объектов может быть исследован на видимость элементов с применением более трудоемкой *проверки по точкам*. Кривую разбивают на мелкие сегменты. На сегменте устанавливают контрольную точку, для которой видимость определяется проверкой: находится ли между этой точкой и центром наблюдения другая принадлежащая объекту точка. Результат переносится на весь сегмент. Для определения видимости точки разработан ряд методов, но все они дают удовлетворительный результат только при очень большом расходе машинного времени. Если проверка по поверхностям вызывает увеличение времени, необходимого для представления полиэдра, приблизительно в 2—5 раз, то проверка по точкам — в 100—1000 раз.

Изображение на рис. 31 получено с помощью одного из алгоритмов удаления скрытых линий.

Внешний вид поверхности зависит от источника света, которым она освещается, ее свойств (цвет, фактура, отражательная способность), а также от ее положения и ориентации относительно источника света и других поверхностей. Лучи света освещают различные участки поверхности объекта неравномерно. Освещенность зависит от того, под каким углом эта поверхность находится по отношению к источнику света. Сильнее всего освещается поверхность, на которую падают перпендикулярные лучи. Чем больше наклон лучей по отношению к поверхности, тем слабее она освещена.

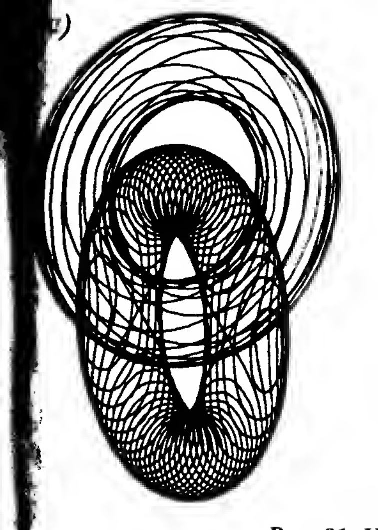
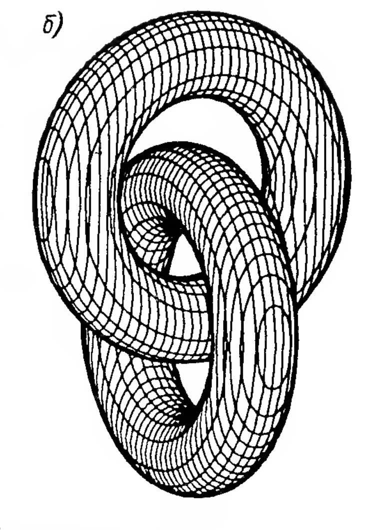



Рис. 31. Изображения:
— включающее невидимые линии; *б* — полученное с помощью алгоритма скрытых линий

Освещенность также уменьшается по мере удаления от источника света.

Чтобы сделать изображение более реалистичным, можно попытаться воспроизвести тени, которые отбрасывает объект, освещенный точечным источником света.

Предметы, освещенные одним точечным источником света, выглядят контрастными. Этот эффект аналогичен тому, который можно наблюдать, когда предмет, помещенный в темную комнату, виден при свете направленной на него вспышки. Все освещенные предметы, в свою очередь, отражают от своей поверхности световые лучи, которые падают на затемненные места, освещая их. Такой отраженный от других поверхностей свет называется *рефлексом*. Вместе с отраженными лучами предметы передают также и свой цвет. Таким образом, даже если предмет защищен от прямых лучей, исходящих от точечного источника света, он все равно будет виден из-за рассеянного света.

Законы светотени познавались человечеством параллельно с законами перспективы. Греческий живописец Аполлодор, получивший от современников прозвище "скиограф", т.е. живописец теней, первый начал изобра-

жать человеческое тело не плоским, а реальным, объемным, пользуясь переходами света и тени. Леонардо да Винчи благодаря мастерству и тонкому использованию светотеневых нюансов достиг исключительного совершенства в передаче формы.

Художники, используя в своей работе законы светотени, прибегают только к качественной оценке: немного светлее (темнее), чуть ярче и т.п. Чем тоньше они чувствуют различия полутонов и оттенков, тем выше реалистический эффект.

Компьютер в отличие от человека плохо "понимает" качественную сторону явлений, но зато гораздо более человека приспособлен к количественной оценке. Чтобы научить компьютер "чувствовать" все оттенки и нюансы свето- и цветопередачи, необходимо перевести физические законы распространения и отражения света на язык машинных команд и дать компьютеру задание рассчитать интенсивность отраженного света в каждой точке наблюдаемого объекта (реального или "придуманного", представляющего результат работы программы).

Так, матовые поверхности обладают свойством *диффузного отражения*, т.е. они рассеивают свет равномерно во всех направлениях. Поэтому кажется, что все участки поверхности освещены одинаково независимо от угла обзора. Для таких поверхностей справедлив закон косинусов Ламберта, устанавливающий соответствие между количеством отраженного света и косинусом угла между направлением на точечный источник света и нормалью к поверхности.

Чем более детально рассматривается характер отражения света от поверхности, тем сложнее математическая модель, но тем и реалистичнее достигаемый с ее помощью эффект.

Взгляните на рис. 32. Изображения, которые вы на нем видите, получены с помощью специальной модели, представляющей поверхность как совокупность микроскопических граней. Каждая из граней — идеальный отражатель. Ориентация каждой грани относительно источника света случайна. Модель построена в предположении, что свет исходит из бесконечно удаленного источника, а следовательно, лучи параллельны.

Напрашивается мысль о следующей возможности практического применения подобной модели отражения. Художник (скульптор) вводит в компьютер не-

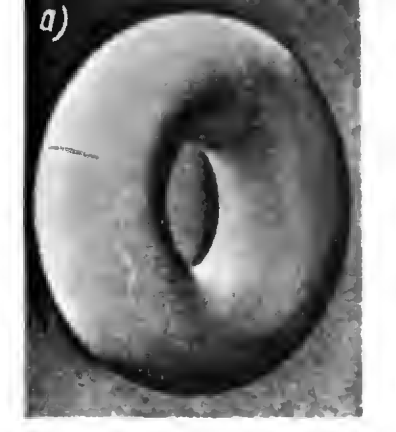




Рис. 32. Тоновые изображения, построенные с учетом особенностей отражения света: от матовой (*а*) и глянцевой (*б*) поверхностей

которое трехмерное изображение — скульптуру, барельеф, либо просто плиту с текстом. Он хочет представить, как его произведение будет выглядеть в гипсе, мраморе, граните, дереве, стекле, бронзе и т.д. Для этого ему достаточно ввести в компьютер конкретные значения коэффициентов отражения. Таким образом, скульптор получит возможность увидеть свою скульптуру еще до того, как она будет изготовлена, и тем самым уберечь себя от непоправимых ошибок.

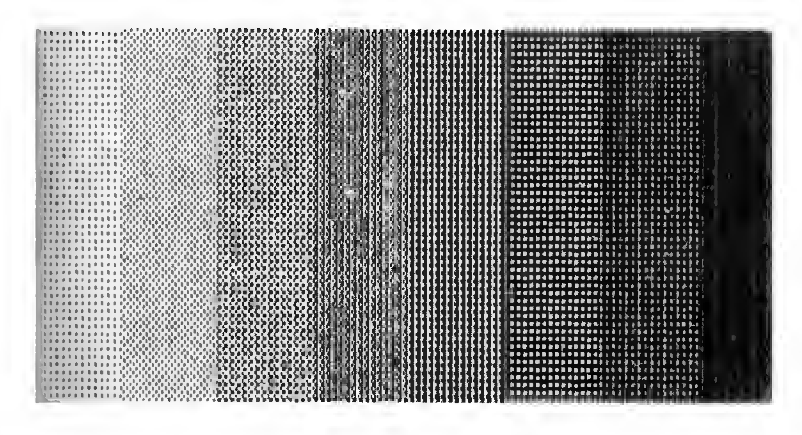


Рис. 33. Шкала тонов



Рис. 34. Портрет, изготовленный на принтере

Тона под лупой. Степень светлоты или темноты какого-либо участка поверхности изображаемого предмета называется *тоном*. *Шкалою* тонов (от лат. scala — лестница) называют последовательный набор тонов, например белый — серый — черный (рис. 33).

Как же получить тоновое изображение с помощью графических устройств?

На экране дисплея тоновое изображение можно получить, регулируя определенным образом яркость луча. Это изображение можно сфотографировать. Но такой путь слишком долог да и дорог.

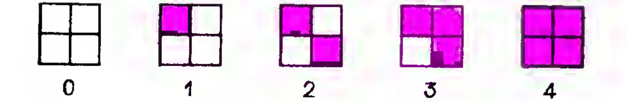


Рис. 35. Пятиградационный элемент печати

Первые "произведения" машинной графики были выполнены с помощью алфавитно-цифрового печатающего устройства, поскольку оно в то время было единственным устройством ввода—вывода. Изобретательные авторы использовали то обстоятельство, что из-за особенностей начертания некоторые символы выглядят темнее, чем другие. Тщательно подбирая знаки, они добивались, чтобы с некоторого расстояния распечатка казалась тоновым изображением, т.е. чтобы знаки сливались (рис. 34). Однако недостаточная интенсивность черного цвета, большие интервалы между соседними строками и между знаками — все это ограничивало число градаций и снижало качество изображений.

Растровый принтер позволил значительно повысить разрешающую способность печати. Но как на растровом принтере получить тоновое изображение, если шкала тонов включает только две градации: черный (точка есть) и белый (ее нет)? Здесь можно применить идею, используемую в полиграфии. Возьмите лупу и внимательно рассмотрите любое тоновое изображение в газете. Вы увидите, что оно составлено из отдельных точек различного размера — где изображение темнее, там размер точек больше, и наоборот. Чтобы реализовать подобный метод при компьютерной печати, выделим *поле* (элемент изображения) размером 2X2 точки. Тогда в за-



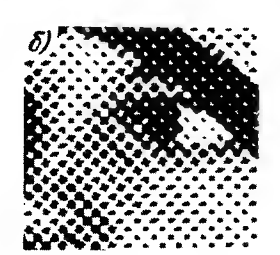


Рис. 36. Тоновое изображение, полученное на растровом принтере:
а — **общий вид;** *б* — **увеличенный фрагмент**

висимости от степени заполнения элемента можно получить 5 градаций тона (рис. 35). Чтобы увеличить число градаций до 16, нужен элемент размером уже 4Х4 (рис. 36).

Причем лучшее качество изображений обеспечивают элементы, исходные точки которых имеют не круглую, а квадратную форму.

Аналогичным образом получают цветное изображение. Так, если на элемент поля размером 2Х2 (рис. 35) выделено всего лишь 3 бита, т.е. по одному биту на красный, зеленый и синий цвета, то можно получить 125 различных цветов. Поскольку каждая конфигурация позволяет выводить на экран до 5 уровней интенсивности для каждого цвета, то это дает $5^3 = 125$ комбинаций цветов.

4. РАБОТА С КОМПЬЮТЕРОМ

Язык программирования является средством связи между программистом и компьютером. Заглянем в "кухню" машинной графики и посмотрим, как можно "приготовить" графическую программу.

"Приготовление" программы — тяжелый и кропотливый труд. Взаимопонимание между людьми зависит от общности моделей мира, которые выработаны образованием, общением, производственной практикой. Чем ближе эти модели, чем больше они похожи, тем легче люди находят общий язык. И наоборот — чем сильнее различаются культуры, тем труднее людям понять друг друга.

Человек и компьютер обладают совершенно различными моделями мира. Человеческому мышлению свойственны образность, многозначность, приблизительность. Информация, вводимая в компьютер, должна быть конкретной, точной, подчиненной жесткой логике. Такая понятная для нас фраза "Пойди, погуляй немного" ничего не говорит компьютеру. Ему нужно конкретно указать, куда идти, что делать и когда вернуться.

Как известно, наиболее употребительные слова являются и самыми многозначными. Поскольку компьютер не может "догадаться", какое значение слова из множества возможных вы имеете в виду, всем словам, зарезервированным в языках программирования,

присваивают одно, заранее оговоренное значение. Исключается употребление синонимов. Строго регламентирован порядок расположения слов в команде.

Как написать программу? Итак, вы решили построить некоторое изображение с помощью компьютера. Чтобы написать программу, вам необходимо:

точно установить, что же должно быть сделано, и определить класс изображений, которые могут быть построены с помощью данной программы, т.е. задать границы ее универсальности;

разработать *алгоритм*, т.е. точно определенную (и конечную) последовательность действий, ведущую к желаемому результату, и выделить в нем функционально однородные части, например: расчет координат, функциональные преобразования, графические команды и т.п.;

закодировать алгоритм, т.е. записать его на одном из языков программирования.

Программа — это подробная последовательность предписаний для компьютера. Если программа состоит из большого числа предписаний, программисты прибегают к специальным приемам. Например, последовательность команд объединяют в группу (модуль), которой присваивают определенное *имя*. Теперь, назвав имя команды, мы предписываем компьютеру выполнить всю последовательность действий. Такой путь называют *модульным программированием*.

Суть модульного программирования можно наглядно проиллюстрировать на следующем примере. Напишем структуру программы построения изображения дома, не углубляясь в детали.

```
СТЕНА: Объект
    переместить перо в [X1, Y1, Z1]
    чертить в [X2, Y2, Z2]
    чертить в [X3, Y3, Z3]
    чертить в [X4, Y4, Z4]
    чертить в [X1, Y1, Z1]
Конец СТЕНА
КРЫША: Объект
. . . . . . . . . . . . . .
. . . . . . . . . . . . . .
Конец КРЫША
ДОМ: Объект
    СТЕНА в [T1]
    СТЕНА в [T2]
```

СТЕНА в [Т3]
СТЕНА в [Т4]
КРЫША в [Т5]
ОКНО в [Т6]
ОКНО в [Т7]
КРЫША в [Т8]
Конец ДОМ

Чтобы построить изображение, мы вызываем подпрограммы (модули) построения стены, крыши и окна. Далее, если мы хотим построить некоторый ансамбль, состоящий из четырех домов, нам достаточно четыре раза обратиться к модулю ДОМ:

ДОМ [В1]
ДОМ [В2]
ДОМ [В3]
ДОМ [В4]

Объединив эти команды в модуль с именем АНСАМБЛЬ, мы можем вызвать всего лишь одну команду, а компьютер построит ансамбль из четырех домов.

Если сравнить процедуру проектирования программы с процедурой строительства домов, то команды можно уподобить кирпичам, модули нижнего уровня — панелям, модули верхнего уровня — блокам. Причем модули, так же как строительные блоки, должны быть унифицированными, чтобы они стыковались (согласовывались входная и выходная информация).

Из чего дом можно построить быстрее? Из кирпичей или из блоков? Конечно, из блоков. Но из блоков возводятся только типовые здания, а для уникальных сооружений требуются кирпичи, которые обеспечивают большое разнообразие архитектурных решений. Но платой за это разнообразие является высокая трудоемкость работ. Так же обстоит дело и в программировании. Иногда очень удобно воспользоваться программным модулем, но они естественно не могут обеспечить выполнение всего разнообразия задач.

Язык, в котором одна команда заменяет множество мелких команд, называют языком *высокого уровня*, на нижнем уровне стоит язык машинных команд.

Управление компьютером можно уподобить управлению таким сложным организмом, как армия. Например, одна команда "Первый батальон — налево" заменяет тысячу команд "Повернуть налево", которые пришлось бы отдавать для каждого солдата батальона. Причем, выполняя эту команду, каждый солдат вызы-

вает из своей памяти последовательность команд (инструкций) более низкого уровня, которая после большой тренировки выполняется им полностью автоматически.

Двух одинаковых программ не существует даже для решения одной задачи. Один и тот же результат может быть получен разными путями, а значит с помощью разных алгоритмов, а один и тот же алгоритм может быть реализован различной последовательностью команд. Так что программирование — это творчество, в программе, как и в произведении искусства, проявляется личность творца. Правда, жизнь заставляет переводить программирование на производственные рельсы, ставить его на конвейер.

С компьютером, а в нашем случае — с графической системой можно работать в пакетном или в интерактивном режимах.

При *пакетном* режиме взаимодействия с графической системой пользователь посылает свою программу на счет в машинный зал, например в виде колоды перфокарт, и ждет результата обработки. Поскольку таких пользователей могут оказаться десятки или сотни, то задача "ждет" своей очереди: она будет просчитана только после всех задач, посланных ранее. Ожидание может продолжаться от нескольких часов до нескольких дней. Если пользователь обнаружит, что он допустил какую-нибудь ошибку, то все повторяется сначала: исправив ошибку в программе, опять приходится "стоять в очереди".

При *интерактивном* взаимодействии пользователь, набрав программу на клавиатуре, тут же наблюдает на экране дисплея за получившимся изображением. Для того чтобы внести в изображение необходимые коррективы, он должен что-то изменить в программе и снова запустить ее на счет. Процесс продолжается до тех пор, пока не будет получено нужное изображение. По сравнению с пакетным режимом на отладку уходит значительно меньше времени. В интерактивной машинной графике рисование и корректировка изображения осуществляются непосредственно на экране.

Как же работать с графической системой?

"Акробатический этюд". "Но причем здесь акробатика?" — скажете вы. И в самом деле, какое отношение имеет акробатика к машинной графике? Термин "акробатика" ассоциируется у нас с ловкостью. Пом-

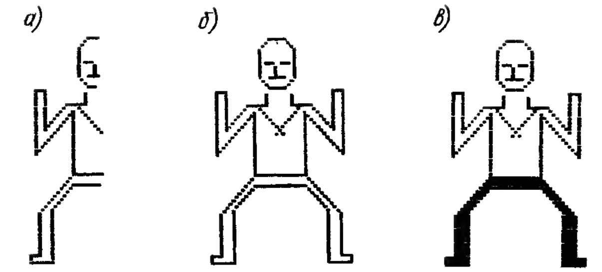


Рис. 37. Акробат Леже

ните, в "Двенадцати стульях" И. Ильфа и Е. Петрова оратор называет журналистов "акробатами пера". Программист, несомненно, должен быть "акробатом терминала". Ловкость манипулирования элементами изображения ему просто необходима.

Попробуем продемонстрировать вам "акробатический этюд" — и на экране дисплея, и на клавиатуре. Для акробатики необходим, как минимум, один акробат. Поскольку устройства графического ввода еще дороги и поставлены на немногие персональные компьютеры, то введем его изображение с клавиатуры в виде последовательности координат точек. Для этого предварительно нарисуем на листе бумаги в клетку стилизованное (упрощенное) изображение.

Измерим координаты (x, y) всех конечных точек отрезков прямых, составляющих контур. Выбрав симметричное изображение, мы можем ввести точки только одной половины, а затем с помощью команды ЗЕРКАЛЬНОЕ ОТОБРАЖЕНИЕ получить другую половину (рис. 37, *а*). При этом объем вводимых координат сокращается почти в два раза. Зеркальное отображение

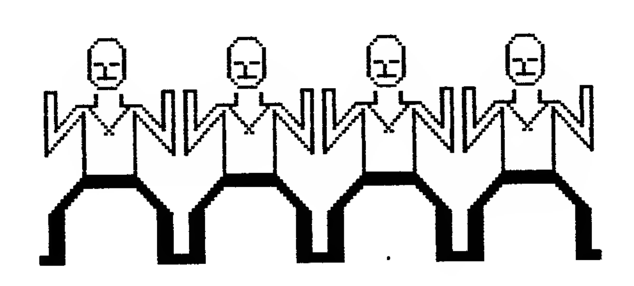


Рис. 38. Братья Леже

получается путем изменения знака координаты x с отрицательного на положительный.

Итак, мы получили последовательность точек, описывающих нашего акробата. Присвоим этой последовательности имя, например Леже. Так как на многих компьютерах клавиатура только с латинскими буквами, то оно будет выглядеть следующим образом: LEGE. Вызывая это имя, мы вызываем из памяти компьютера соответственно и нашего акробата Леже (его изображение).

Итак, акробат готов (рис. 37, *б—в*). Он, может быть, не очень красив, но у него железные руки, крепкие ноги и он необыкновенно ловок. В этом вы сейчас убедитесь!

На арену цирка выходит конферансье и торжественно объявляет: "Впервые на ар-рене цирка! ... Акробаты братья Леже!". Оркестр играет марш, и на арену выбегают четыре брата-близнеца.

Пожалуйста! Что может быть для компьютера проще, чем генерировать "близнецов". При каждом вызове имени LEGE генерируется один близнец. Нам достаточно четыре раза вызвать имя LEGE (это удобнее сделать с помощью оператора цикла) и с помощью команды ЧЕРТИТЬ вывести их на экран дисплея в заданные области, т.е. со сдвигом по оси x (рис. 38). Поскольку командой ЧЕРТИТЬ заканчиваются все графические программы, далее мы ее упоминать не будем.

"Следующий номер-р! ... Колонна!". В построении колонны участвуют четыре акробата (рис. 39). Поражает сила рук — действительно, у Леже они железные!

Для построения колонны используются те же команды, что и в предыдущей ситуации — при выходе братьев на арену. Только изображение LEGE выводится со сдвигом по оси Y.

"Стойка на голове!...". Верхний акробат разбегается, подпрыгивает и, перевернувшись в воздухе на пол-оборота, делает стойку на голове (рис. 40).

Здесь в ход идет команда ВРАЩЕНИЕ на 180°.

"Силовой номер! Мельница-а!" (рис. 41). Один держит троих! Действительно, номер, достойный Дикуля. Как вы можете себе представить, здесь возникает колоссальная нагрузка на голову (не умственная, а физическая). Видимо, в этом причина восьмигранности голов братьев.

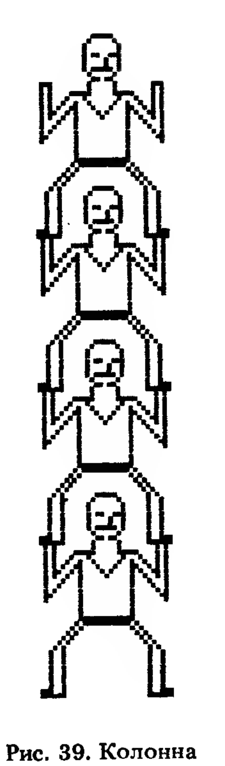


Рис. 39. Колонна

Рис. 40. Стойка на голове

Рис. 41. Мельница

Рис. 42. Пирамида

Использована команда ВРАЩЕНИЕ на 360° с шагом 90°.

"Оригинальная пирамида!" (рис. 42).

Здесь последовательность команд такова: СДВИГ ПО X, СДВИГ ПО Y, ВРАЩЕНИЕ НА 180°.

"Заключительный номер-р!.." Гаснет свет! Оркестр играет туш. На арену выбегают 8 акробатов, загримированных под братьев Леже. В руках они держат бенгальские огни. Серия головокружительных прыжков заканчивается построением сложнейшей акробатической фигуры, которой еще нет названия (рис. 43). Как построить программу для этой фигуры, вы уже, конечно, представляете. А как получились бенгальские огни, мы расскажем далее.

А сейчас поаплодируем братьям и подумаем, что же нам дал этот акробатический этюд? Только за счет двух графических команд — ВРАЩЕНИЕ и СДВИГ — мы смогли выполнить самые разнообразные "номера". Для этого нам не нужно было ни размечать, ни измерять, ни рассчитывать, ни рисовать. Здесь не требуется скрупулезная аккуратность при программировании, которую диктует пакетный режим, когда результаты можно получить спустя несколько часов после посылки задачи на счет. В интерактивном режиме допущенная ошибка тут же выявляется компьютером и выводится на дисплей для исправления. Мы можем себе позволить



Рис. 43. Заключительный номер

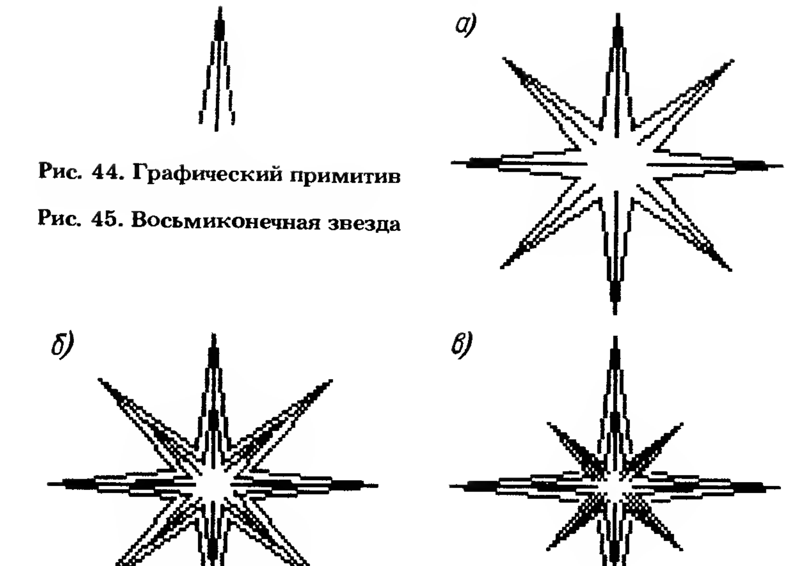


Рис. 44. Графический примитив

Рис. 45. Восьмиконечная звезда

такую вольность, как работа по методу проб и ошибок, поскольку "стоимость" ошибки достаточно низка.

"Звездопад". Вы уже убедились, что даже малыми средствами можно добиться интересных результатов. Еще большие возможности открывает перед нами команда МАСШТАБ. Построим простейший графический элемент, или примитив (рис. 44), набрав на клавиатуре координаты четырех конечных точек трех сходящихся отрезков. Вызвав команду ВРАЩАТЬ, будем поворачивать графический примитив на 360° с шагом 45°. Получим изображение восьмиконечной звезды (рис. 45, *а*). "Закрасим" звезду: заполним ее двумя звездами меньшего масштаба. Варьируя масштаб изображения и число "вкладываемых" звезд, можно изменять степень зачернения звезды (рис. 45, *б*).

Если при вращении графического примитива одновременно уменьшать его масштаб в два раза при повороте на 45, 135, 225 и 315°, можно получить звезду другой формы (рис. 45, *в*).

Теперь вам будет легко разобраться, как построен рисунок "Звездопад" (рис. 46). "Падающие" звезды

меньше исходной в 4 раза. Их количество и сдвиг задаются в цикле.

Три деревца в правом углу получены за счет вложения друг в друга десяти графических примитивов разного масштаба.

Непосредственно рисованием здесь можно назвать только ввод графического примитива. Все остальное — программные манипуляции изображением с помощью клавиатуры, которые быстро выполняются компьютером и отображаются на экране дисплея. В полученное изображение можно внести коррективы, изменив программу и вновь запустив ее .

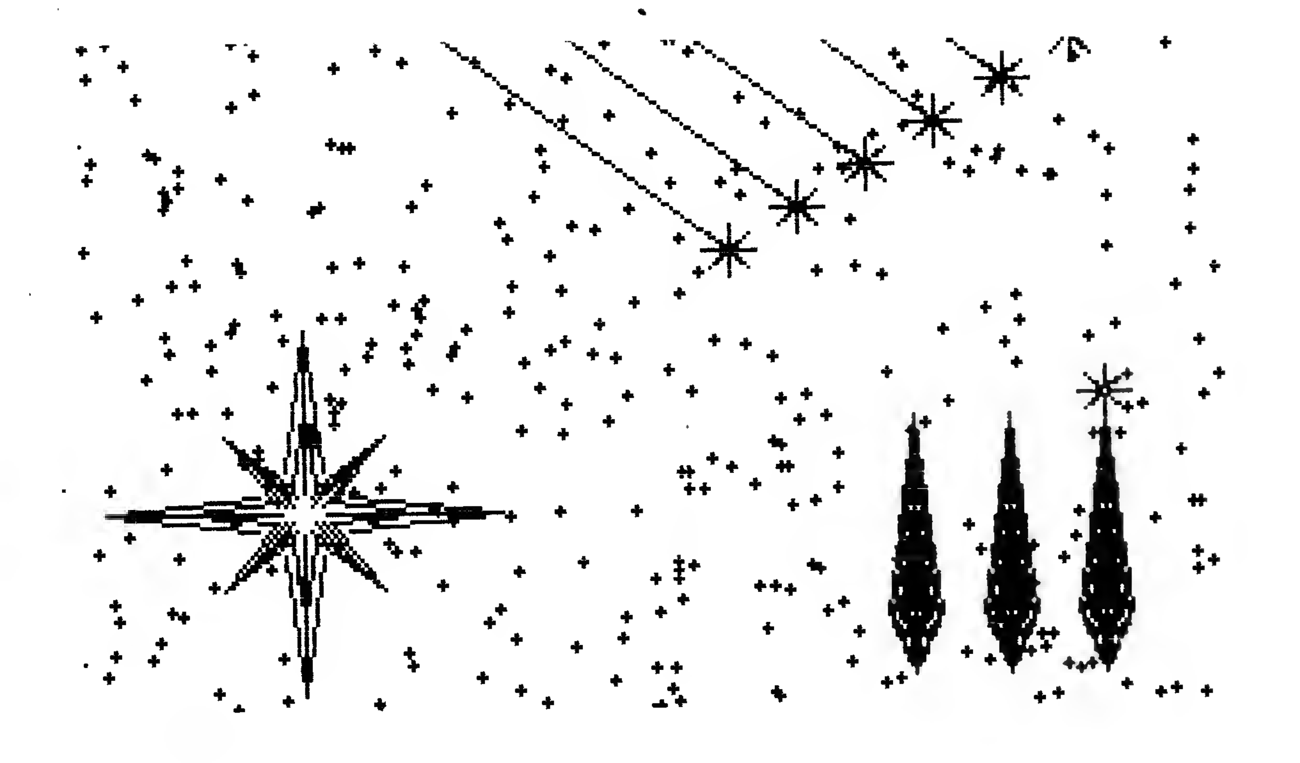
"Меню". Мы рассмотрели вариант общения пользователя с компьютером с помощью команд. Но для такого общения нужно досконально знать все команды — как их содержание (семантику), так и форму (синтаксис). Стоит вместо точки, например, поставить запятую или двоеточие, и команда не будет воспринята компьютером. Изучение всех команд и их запоминание требуют значительного времени и усилий. Чтобы облегчить общение с компьютером, сделать его возможным для необученного пользователя, разработчиками программного обеспечения создается так называемое "*меню*", т.е. набор альтернатив, который предлагает пользователю машина, начиная диалог. Каждой альтернативе соответствует определенный символ (математический знак) или пиктограмма (условный рисунок). Из высвечиваемого на экране дисплея меню пользователь выбирает то, что ему нужно, фиксируя свой выбор набором на клавиатуре числа или мнемонического[1] кода, которые указываются в меню.

Появились сообщения о сенсорных экранах. Чтобы известить машину о своем выборе, достаточно пальцем прикоснуться к определенной области экрана, на которой представлена строка либо ее мнемонический код.

Предположим, вам необхсдимо подготовить для стенгазеты заметку о защите окружающей среды. Для оформления вам необходим заголовок "Берегите природу!" и картинки, например изображения зайца, птицы, растений. Как должно быть организовано меню, чтобы это задание мог выполнить человек, не имеющий специальных навыков?

---

[1] *Мнемонический* — предназначенный для запоминания.


Рис. 46. Звездопад

При первом обращении к меню на экране дисплея высвечиваются строки:

1. ТЕКСТ
2. СИМВОЛЫ
3. ЖИВОТНЫЕ
4. РАСТЕНИЯ

Вам нужна строка 3 "ЖИВОТНЫЕ". Для ее вызова достаточно набрать на клавиатуре число "3", и на экране высветится более подробная информация об изображениях животных, хранящихся в памяти компьютера:

1. БЕСПОЗВОНОЧНЫЕ
2. ПРЕСМЫКАЮЩИЕСЯ
3. МЛЕКОПИТАЮЩИЕ
4. ПТИЦЫ
5. РЫБЫ

Выбрав строку 3, получаем полный перечень всех млекопитающих.

1. Белка 2. Волк 3. Заяц ... 35. Шакал

Снова набираем число "3". На экране появляется стилизованное (упрощенное, чтобы экономить память компьютера) изображение зайца. С помощью функциональной клавиатуры это изображение можно передвигать по экрану, изменять его масштаб и цвет. Аналогичным образом выбираем любое другое изображение.

Чтобы набрать заголовок, возвращаемся, нажав специальную клавишу, в начало меню и указываем на строку 1 "ТЕКСТ". При этом высветится меню шрифтов:

1. ПРЯМОЙ
2. КУРСИВНЫЙ
3. НАКЛОННЫЙ

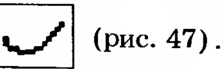
Выбрав нужный шрифт, набираем заголовок "Берегите природу!", перемещаем его по экрану, масштабируем. Вот и все! Теперь можно "снять копию" с экрана, т.е. вывести полученную картинку на принтер. Просто? Но за этой простотой стоит труд программиста, составившего меню, выбравшего из великого множества изображений то немногое, что может быть введено в память компьютера, выполнившего классификацию изображений, их кодирование и ввод. Это очень большая работа. Она целесообразна только в том случае, когда созданное программное обеспечение удовлетворит потребностям многих пользователей.

Естественно, что не все запросы может удовлетворить одно меню. Например, сколько геометрических тел нужно задать и хранить в памяти компьютера, чтобы каждый пользователь мог выбрать фигуру нужных размеров, формы и ориентации?

Конечно, меню значительно облегчает общение с компьютером, но в то же время и ограничивает ваши возможности. Вы можете взять только то, что вам дают, так в магазине самообслуживания вы берете только то количество продукта, которое упаковано. Язык команд позволяет делать почти все, но требует длительного обучения. Поэтому он применяется подготовленными пользователями.

Рисуем на экране. Давайте, как на уроке, повторим пройденное. В этом нам поможет рассказ вчерашнего десятиклассника Кирилла Кудрявцева. В нем вы найдете и то, что вам уже известно, и кое-что новое.

"Я не умею рисовать, но когда мне разрешили работать на персональном компьютере, я решил нарисовать Карлсона. У меня была книжка А. Линдгрен "Малыш и Карлсон", из которой я взял понравившийся мне рисунок. Рисовал с помощью мыши и светового пера, пользуясь методом "резиновой нити": "нить" между двумя точками "натягивают", катая мышь. (Сначала мне было нелегко добиться, чтобы курсор передвигался в нужную точку, но потом появился навык и дело пошло быстрей.) Чтобы из прямой линии сделать кривую, нужно световым пером "подцепить" отрезок примерно посередине и "оттянуть" его.

Чтобы вызвать программу рисования "резиновой нитью", я нашел в меню, которое высвечивается на экране слева, знак  (рис. 47).

Затем я решил раскрасить рубашку. Для этого из меню вызвал команду закраски — знак . Внизу экрана высветилось меню закраски. Я выбрал для рубахи Карлсона ткань в клеточку и указал световым пером на точку внутри закрашиваемой области. Поскольку лямка, на которой у Карлсона держатся брюки, разделяет закрашиваемую область, то я указал на точку в области и по другую сторону от лямки. Рубашка готова. Так же закрасил башмаки (рис. 48).

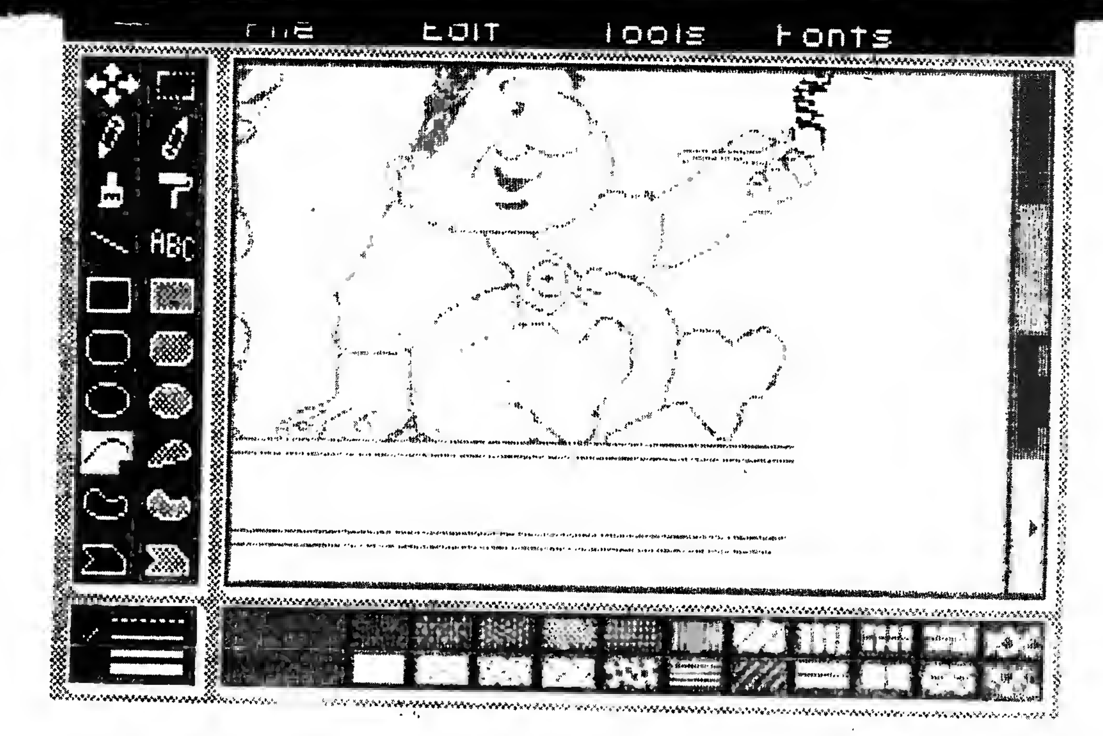


Рис. 47. Контуриый рисунок

Рнс. 48. "Раскрашенное" изображение

Рнс. 49. Законченный рисунок

Чтобы не возникло сомнений в том, что это Карлсон, рисунок я подписал с помощью программы написания текстов НВС. Сначала я задал две противоположные (диагональные) вершины прямоугольника — область текста, набрал на клавиатуре слово "КАРЛСОН", и на экране появилась надпись заданного размера. Я убрал с экрана меню, и вот мой рисунок готов (рис. 49)".

5. КОМПЬЮТЕР В КОНСТРУКТОРСКОМ БЮРО

В Советском Союзе работает свыше 2,5 млн конструкторов и проектантов. Что только им не приходится проектировать — крупные заводы и атомные электростанции, орбитальные станции и суперкомпьютеры. Современные приборы и технические системы настолько сложны, что традиционные, "ручные" методы проектирования уже не удовлетворяют сегодняшним потребностям, не могут обеспечить необходимую производительность этого процесса и одновременно гарантировать высокое качество проектов. Возьмем, к примеру, проблему проектирования сверхбольших интегральных схем (СБИС), которые содержат до 400 тыс. транзисторов. Все элементы СБИС нужно расположить таким образом, чтобы обеспечить минимальную длину проводников, при этом проводники не должны пересекаться.

Но правильно спроектировать объект еще недостаточно. Его надо спроектировать быстро, иначе к моменту выпуска он морально устареет. Причем в наш век сроки морального старения неумолимо сокращаются.

Вместе с тем с усложнением проектов усугубляются и экономические последствия неверных проектных решений и ошибок, вероятность которых при традиционном способе проектирования довольно высока в силу крайне ограниченных способностей человека к переработке информации. Устранение таких ошибок на этапе эксплуатации созданной системы стоит примерно в тысячу раз дороже, чем на этапе ее проектирования. Поэтому столь важное значение приобретает сейчас проблема автоматизации проектно-конструкторских работ.

Необходимо освободить проектанта от трудоемких, многократно повторяющихся процессов для творчес-

кой работы. Общение пользователя с системой автоматизации должно быть простым и естественным, а время выполнения команды — достаточно малым. Поэтому языки конструирования разрабатываются таким образом, чтобы работа с компьютером не требовала специальной подготовки.

Компьютер позволяет перейти к "безбумажной" технологии проектирования, поскольку документация хранится на машинных носителях.

В то же время не будем закрывать глаза на "оборотную сторону медали". Как показал опыт, переход от бумаги, карандаша и кульмана к "виртуальной" (кажущейся) чертежной поверхности, световому перу и графопостроителю, от натурного моделирования узлов и каскадов аппаратуры к машинному, от справочников и каталогов к библиотекам и банкам данных является процессом весьма трудным.

Сложные структуры команд и обилие несовместимых программных средств заставляют некоторых разработчиков с тоской вспоминать о тех днях, когда проектные работы велись при помощи карандаша, ластика и бумаги в клетку. Чтобы добиться при работе с АРМ того же мастерства, которым разработчики обладали при ручном проектировании, часто требуется не менее года.

Однако необходимость автоматизации проектирования очевидна.

Что такое АРМ? *Автоматизированное рабочее место* (АРМ) представляет собой графический вычислительный комплекс, предназначенный для автоматизации проектирования. Иногда АРМ называют рабочей станцией. Стоимость подобных систем довольно высока, они отличаются высокой разрешающей способностью, большими объемами памяти и высоким быстродействием.

В настоящее время в СССР выпускаются три типа АРМ: конструктора радиоэлектронной аппаратуры (АРМ-Р), конструктора изделий машиностроения (АРМ-М) и программиста станков с числовым программным управлением (ЧПУ). Эти комплексы построены на базе малых ЭВМ. В состав периферийного оборудования входят накопители на магнитных дисках, клавиатура, полуавтоматические кодировщики графической информации, графические дисплеи и графопостроители. Программное обеспечение позволяет вводить в машину

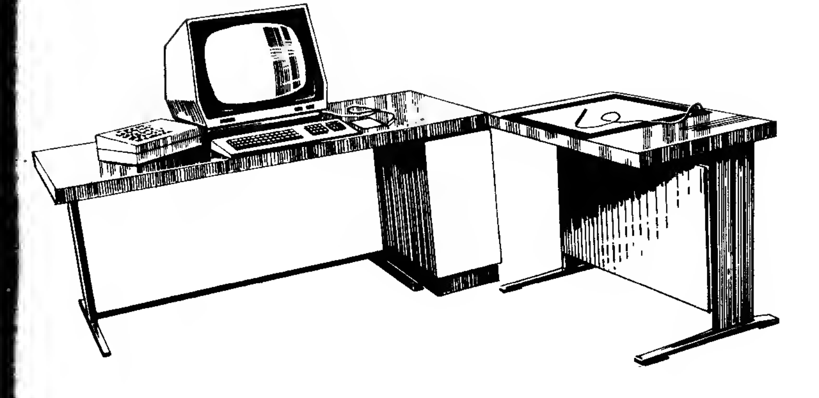


Рис. 50. Автоматизированиое рабочее место

и рассматривать любые фрагменты чертежей в увеличенном масштабе, корректировать чертежи, создавать архивы их описаний, вводить и редактировать текстовые материалы, получать окончательные результаты проектирования в виде чертежей и распечаток текста, подготавливать перфоленты для управления станками с ЧПУ.

На рис. 50 показано АРМ, выполненное на базе ЕС ЭВМ. К одной ЭВМ может подключаться до четырех АРМ.

Развитие графических АРМ началось с конфигураций с 16-битовой мини-ЭВМ и одним графическим терминалом. Сначала их применение ограничивалось проектированием печатных плат для электронных устройств. Низкая производительность таких систем привела к идее распределения графических задач между несколькими независимыми процессорами. Наличие нескольких процессоров позволяет параллельно выполнять несколько процессов и тем самым сделать работу графической системы более эффективной.

Наиболее распространены графические системы с распределенным программным обеспечением, включающие два процессора — центральную и абонентскую ЭВМ, связанные каналом передачи данных (рис. 51).

Абонентская ЭВМ входит в состав рабочего места пользователя графического программного обеспечения. Ее вычислительные ресурсы (объем памяти, быстродействие) ограничены. В большинстве случаев это —

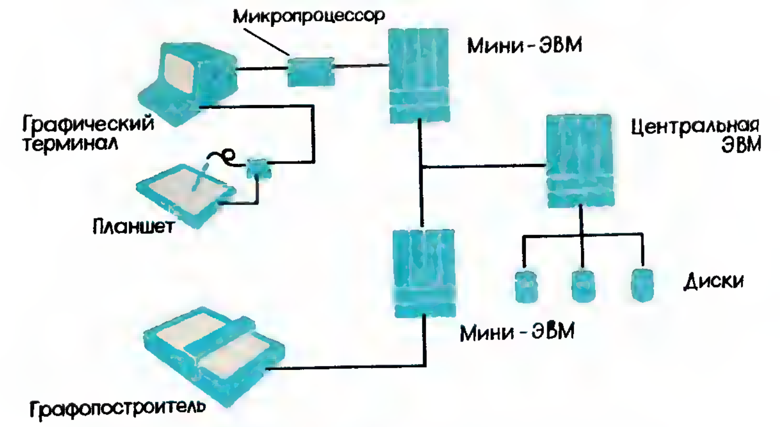


Рис. 51. Графическая система с двумя процессорами

мини- или микроЭВМ. Доступ к графическим программам осуществляется абонентом с помощью графического оборудования, имеющегося в составе этой ЭВМ.

Центральная ЭВМ — удаленная от абонентских рабочих мест высокопроизводительная машина, обладающая большими объемами внешней памяти (десятки, сотни мегабайт) на магнитных дисках. Основной объем работ осуществляется на периферийных машинах, центральная ЭВМ предназначена для обмена данными, управления выполнением отдельных работ и выполнения больших программ.

Такая организация графической системы позволяет за счет использования абонентской ЭВМ освободить центральную ЭВМ от обработки мелких запросов, обеспечить быстрый отклик системы на запросы, а в случае необходимости — доступ к ресурсам мощной центральной ЭВМ.

Работа с призраком. Конструктор воплощает свои подчас неясные идеи в материальную форму через информационную модель — техническую документацию. По чертежам не всегда легко представить себе проектируемый объект, велика вероятность ошибки, поэтому приходится изготовлять макеты — макеты заводов, улиц или городов, макеты самолетов и т.п. Это трудоемкий и долгий процесс, к тому же и макеты не всегда наглядны. Приходится изготовлять дополнительные разрезы, сечения. Кроме того, практически невозможно получить действующий макет объекта.

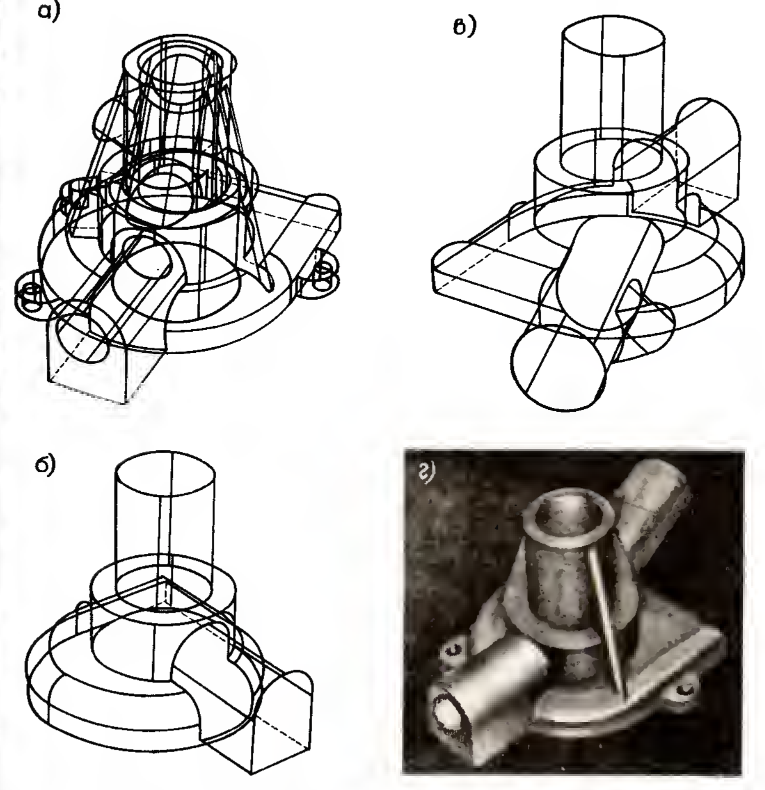


Рис. 52. Построение изображения детали сложной конфигурации на экране дисплея:
а – *в* — **контурного;** *г* — **тонового**

И здесь как нельзя кстати компьютер. Он помогает увидеть то, чего еще нет, — "призрак" детали (рис. 52, 53). Причем этот "призрак" является полной информационной моделью детали, его можно "распилить" по любой из плоскостей, узнать его массу, задав материал детали, соединить в сборке с другими подобными деталями, проверив их сопрягаемость.

Взгляните на рис. 54. На нем изображен так называемый "невозможный объект". Подобные рисунки доказывают нам, что не всегда можно верить даже собственным глазам. Теперь представьте себе такую ситуацию. Проектируется сложнейший современный промышленный комплекс. Получены кипы технической

документации. А при строительстве оказывается, что на рабочем чертеже от агрегата отходят четыре трубы, а к другому агрегату подводятся только три! Избежать таких ситуаций поможет дотошный "контролер", память которого хранит точные сведения о каждом блоке, который скрупулезно проверяет согласованность всех соединений, стыков, разъемов. Вы, конечно, уже догадались, что речь идет о компьютере.

Еще большие возможности открывает "оживление" изображения на экране дисплея. Предположим, разработчик автомобиля должен выбрать такое расположение рулевой колонки, чтобы обеспечить безопасность водителя при столкновении. Раньше для этого проводили так называемые натурные испытания, т.е. в автомобиль усаживали манекен (которого испытатели любовно называют Иван Иванычем) и закрепляли на нем множество датчиков. После этого разгоняли автомобиль до определенной скорости и направляли в бетонную стенку. При этом производилась киносъемка столкновения и регистрировались показания датчиков. Подобный метод испытаний, во-первых, очень дорог, поскольку при каждом испытании разбивается автомобиль. Во-вторых, очень много времени занимает подготовка эксперимента и обработка полученных данных.

Возможности компьютерной техники позволяют отказаться от натурных испытаний. Достаточно заложить в память компьютера математическую модель движения инерционного тела при столкновении. Конструктор вводит данные о скорости автомобиля, жесткости кузова и преграды, "сажает" в салон автомобиля модель — Иван Иванычa (рис. 55, 56).

Существует большое число моделей, описывающих человеческое тело.



Рис. 53. Цветное изображение изделия на экране дисплея

Рис. 54. Невозможный объект

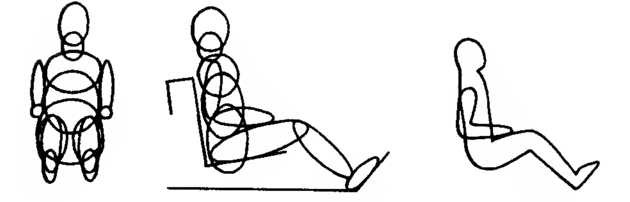


Рис. 55. Графическая модель водителя автомобиля

Например, модель можно составить из эллипсоидов (рис. 57, *б—г*). Каждый эллипсоид представляют девятью числами: три числа — длины полуосей, три числа — координаты *x*, *y*, *z* и еще три числа — углы, определяющие ориентацию осей. Более груба трехмерная графическая модель, синтезированная из параллелепипедов (рис. 57, *а*). Подобные модели используются все чаще; их сажают в кабины самолетов и в космические аппараты; подчиняясь командам тренера, балетмейстера, они ходят, прыгают, танцуют, кувыркаются на экране дисплея.

Но вернемся к нашему случаю. Подготовив все условия для аварии, конструктор-программист запускает программу столкновения. Теперь дело за компьютером. Он быстро выполняет множество расчетов, и на экране графического дисплея воспроизводит движение тела водителя при аварии.

Предположим, что водитель ударяется о рулевую стойку. Конструктор отодвигает ее немного вперед и повторяет аварию. Это продолжается до тех пор, пока не будет найдено оптимальное расположение стойки в салоне. Причем может быть воспроизведен любой момент аварии. Автомобиля еще нет, нет даже его натур-

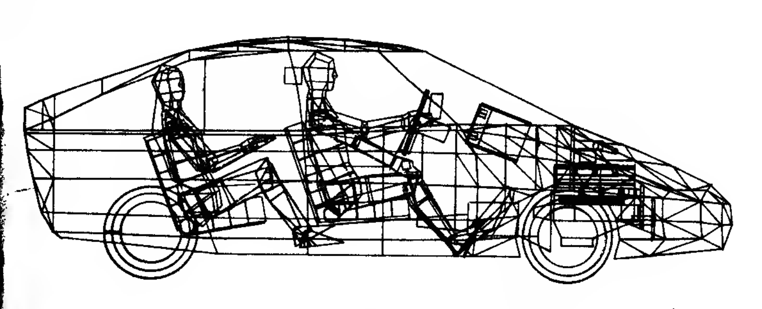


Рис. 56. Проектирование автомобиля

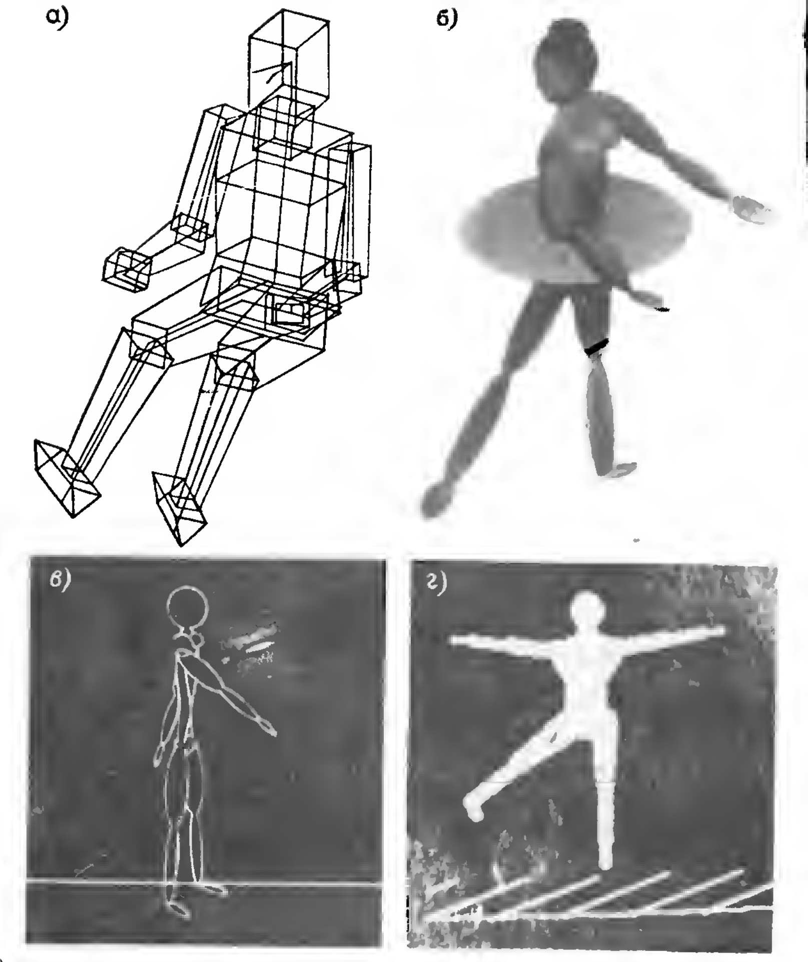


Рис. 57. Графические модели тела человека

ной модели, но конструктор уже может оценить его достоинства и недостатки, ''протрясти на ухабах'', провести по любому покрытию, с любой скоростью, под различным уклоном.

Моделирование поведения проектируемого объекта в предполагаемых обстоятельствах может использоваться во многих областях: для проверки работы спроектированной электронной схемы в различных режимах, траектории движения несуществующих узлов несуществующего станка и т.д.

Проблемно-ориентированные языки проектирования. Когда в нашей стране началось широкое внедрение вычислительной техники в проектные работы, перед разработчиками лингвистического (языкового) обеспечения встала задача создания проблемно-ориентированных языков, т.е. рассчитанных на решение определенной проблемы, некоторой категории задач. В гл. 2 мы рассмотрели некоторые команды проблемно-ориентированных языков, созданных на базе английского языка. Но лингвистическое обеспечение систем автоматизированного проектирования нужно было строить так, чтобы команды были построены на основе русского языка и были бы понятны проектанту после небольшой подготовки.

Созданы языки графического взаимодействия пользователя с системой автоматизированного проектирования с помощью команд. Эти команды похожи на обычные предложения русского языка и содержат глаголы, существительные, модификаторы и знаки пунктуации. Языки позволяют формировать описания чертежей, деталей для обработки на станках с ЧПУ, принципиальных электрических схем, печатных плат и другую графическую и текстовую информацию.

Язык является композицией легко запоминаемых командных мнемоник. Предложение (команда) в общем случае состоит из глагола, существительного, модификатора и координатных данных. Например: # СТР ЛИН — стереть линию (СТР — глагол, ЛИН — существительное); # ВВС ЗАК ПДЧ R 10 — ввести закругление радиусом 10 мм с подчисткой выступающих линий (ВВС — глагол, ЗАК — существительное, ПДЧ и R10 — модификаторы). Знак # означает готовность диалоговой системы принять команды.

Проектант осуществляет ввод предложений в виде директив с помощью средств графического взаимодействия (клавиатуры, кодировочного планшета, светового пера). Диалоговая система немедленно интерпретирует команды и отображает результаты их выполнения на графическом дисплее.

Поскольку АРМ предназначаются для работы с пользователями, не искушенными в программировании, то очень важно обеспечить максимальную простоту взаимодействия с графической системой. С этой точки зрения очень удобно меню.

Выбрав в меню необходимую команду, пользователь указывает на соответствующую ей мнемонику световым пером. Поскольку пользователь должен быть уверен, что команда воспринята правильно, необходимы специальные способы индикации. Одним из распространенных приемов является следующий: световая кнопка, которая указана (воспринята) последней, начинает мерцать. Таким путем пользователю сообщается конкретное состояние, в котором находится система.

Чтобы дать знать системе, с каким элементом изображения будет проводиться манипуляция, достаточно указать на любую его точку световым пером. Возникает вопрос: как в насыщенном элементами изображении машина распознает именно тот, который вам нужен? Ведь точка, на которую пользователь указывает световым пером, принадлежит отрезку, отрезок может входит в объект, например, ОКНО, ОКНО, в свою очередь, может входить в объект ДОМ и т.д. Дело в том, что в системе предусмотрены две команды: ПОДНЯТЬСЯ ВВЕРХ ПО ИЕРАРХИИ и ВЕРНУТЬСЯ ВНИЗ ПО ИЕРАРХИИ. Когда вы указываете световым пером на некоторый злемент, система повышает яркость объекта, содержащего выбранный элемент и относящегося к самому нижнему уровню иерархии. Если вам нужен более крупный объект, введите команду ПОДНЯТЬСЯ ВВЕРХ ПО ИЕРАРХИИ. При этом подсвечивается объект следующего по иерархии уровня, содержащий выбранный элемент. Если и зтот объект не нужен пользователю, команда повторяется и подсвечивается еще более крупная часть изображения. Если же вы по ошибке подниметесь слишком высоко, то можно исправить положение с помощью команды ВЕРНУТЬСЯ ВНИЗ ПО ИЕРАРХИИ.

Но и зто не все. Сейчас ученые бьются над тем, чтобы управлять графической системой с помощью речевых команд. В настоящее время уже функционирует ряд устройств, служащих для распознавания набора отдельных слов, четко произносимых диктором. Максимальная емкость словаря предусматривает возможность записи слов длительностью до 2,6 с, а одна фраза может состоять из 21 связного слова. Устройства реализованы на основе однокристалльной микроЭВМ. В некоторых системах проектирования 50 слов позво-

ляют выполнять около 80% операций, реализуемых с терминала.

Одно произнесенное слово может заменить до 40 нажатий клавиш, так что пользователю больше не приходится запоминать сложные последовательности набора команд на клавиатуре. Как показала практика, использование устройств речевого ввода позволяет повысить эффективность систем автоматизированного проектирования на 40—200%.

6. КОМПЬЮТЕР В ДИЗАЙНЕ

Вы, наверное, уже убедились, что компьютер может оказать неоценимую помощь при проектировании технических объектов и систем. А есть ли место компьютеру в художественном конструировании — дизайне?

Работа дизайнера представляет собой попытки материализовать свою идею в форме рисунка, пластилиновой или другой модели. Немало вариантов переберет дизайнер, прежде чем примет окончательное решение. Поиск может быть не только долгим и мучительным, но и трудоемким.

Ускорить и облегчить материализацию идеи, ее модификацию поможет компьютер. Как и конструктор, дизайнер сможет на экране дисплея синтезировать реалистическое изображение проектируемого изделия из множества элементов-заготовок, увидеть, как оно будет выглядеть в разном исполнении, варьировать цветовое решение, совершенствовать форму. Возможность быстрого просмотра множества вариантов позволит повысить эстетические качества модели.

Представьте себе: художник-модельер световым пером рисует на экране дисплея фигуру девушки, которую затем "одевает" (в платье, костюм, пальто). После этого с клавиатуры "задает" ткань — цвет и рисунок (рис. 58), добавляет или убирает карманы, пояс, подбирает отделку, украшения, фурнитуру, варьирует цветовую гамму. Работа идет быстро и очень увлекательно, идеи художника реализуются практически мгновенно.

Орнамент. Слово "орнамент" произошло от латинского ornamentum (украшение). Оно обозначает узор, характеризующийся ритмическим чередованием элементов — геометрических или стилизованных растительных и животных мотивов.

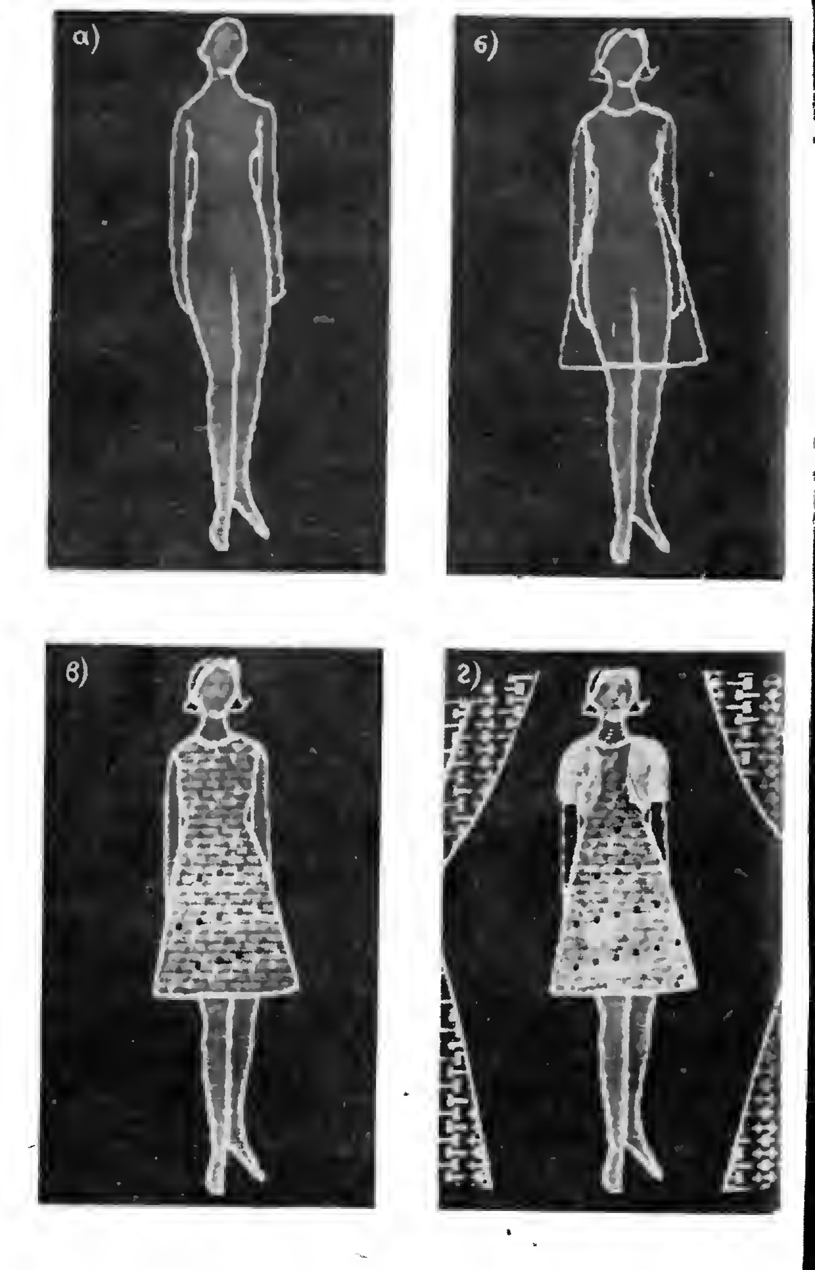


Рис. 58. Моделирование одежды на экране дисплея

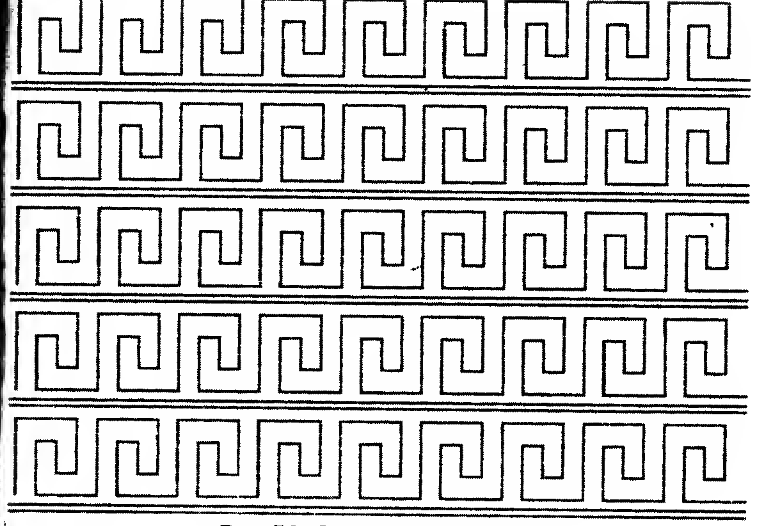


Рис. 59. Орнамент "меандр"

Возникнув в глубокой древности [например, распространенный геометрический орнамент "меандр" (рис. 59) применялся еще в Древней Греции, свое название он получил от извилистой реки Меандр], орнамент прочно вошел в наш быт. Посмотрите вокруг: вы найдете его в рисунке обоев, на тканях, коврах, декоративных предметах, стенах зданий.

Отрезки прямой линии и простые геометрические фигуры (прямоугольник и окружность), организованные определенным образом, создали неповторимый рисунок решетки Летнего сада, вызывающей восхищение во всем мире (рис. 60).

Ритмические повторения рисунка, лежащие в основе орнамента, обусловлены повторением элементов трудовых процессов (в ткачестве, плетении, строительстве и т.д.). Переход от кустарного производства к промышленному сопровождался дальнейшим распространением орнамента, но при этом характер работы художника остался прежним. Ему приходится тщательно прорисовывать или вычерчивать все детали орнамента, затем их раскрашивать. При этом, чтобы посмотреть, как будет выглядеть орнамент в другой цветовой гамме, всю работу необходимо повторить. Это очень кропотливый и однообразный труд. Какие же возможности предоставляет художнику компьютер?

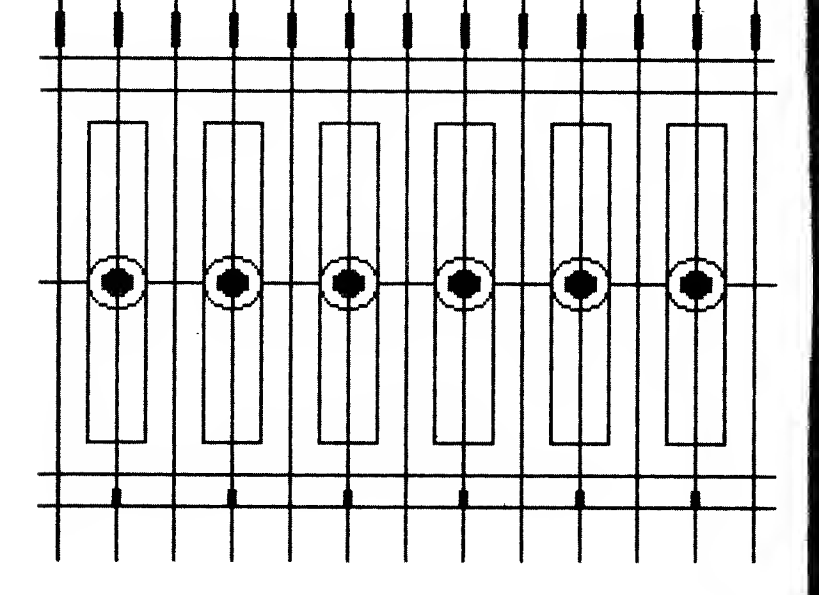


Рис. 60. Решетка Летнего сада на компьютере

Компьютер может быть использован как *инструмент для рисования.* Повторяющийся элемент орнамента можно синтезировать на зкране дисплея, а затем формировать орнамент, пользуясь всеми рассмотренными выше приемами.

Даже используя такую простейшую геометрическую фигуру, как квадрат, можно получить разнообразные орнаменты. Зададим число квадратов внутри элемента орнамента, число элементов по осям x и y, составим простенькую программу. Варьируя только одну величину — шаг изменения стороны квадрата, мы можем получить орнаменты, которые вызывают различные зрительные эффекты (рис. 61).

При создании орнаментов могут широко использоваться различные оптические эффекты. Так, орнамент на рис. 62, *а* состоит только из черных линий. Но поскольку черные линии воспринимаются нами как тени, мы "видим" и белую линию (границу рельефа), от которой должны были бы падать тени, но которой на самом деле нет ! Знание особенностей человеческого зрения позволяет добиваться интересных результатов (рис. 62, *б*).

а)

б)

Рис. 61. Вариации на тему "квадрат"

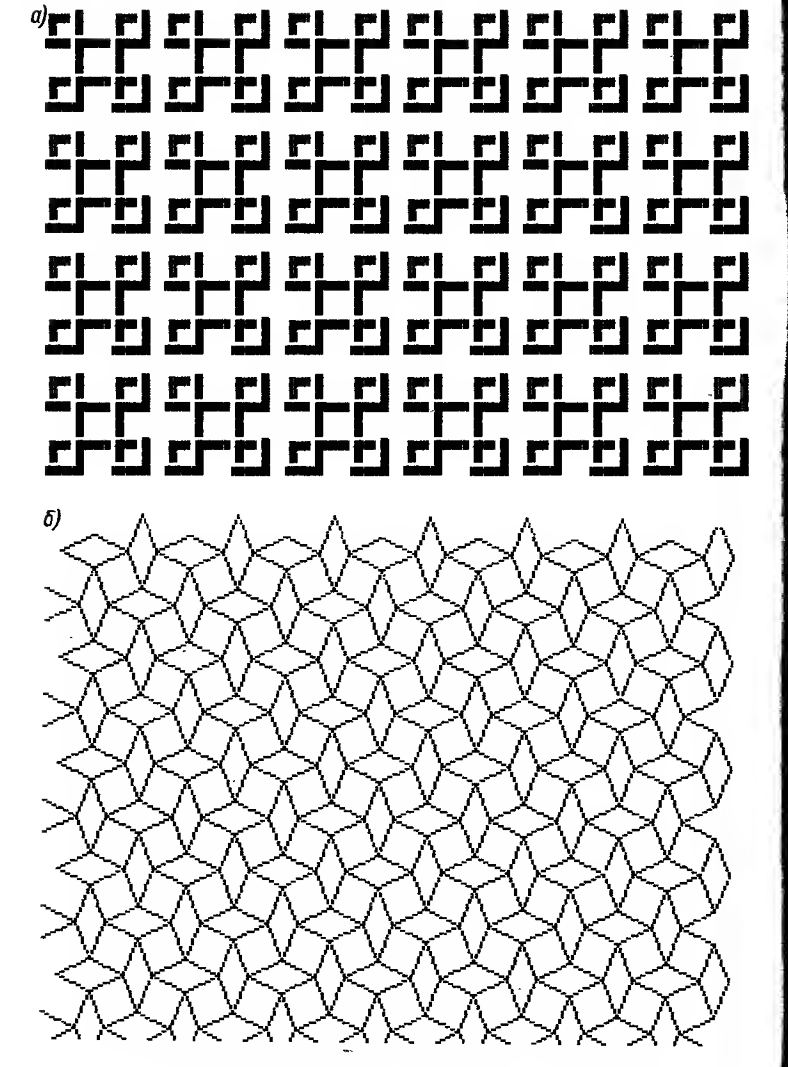


Рис. 62. Орнаменты с визуальными эффектами

Задав на клавиатуре номер цвета для каждого элемента орнамента и фона, вы тут же можете оценить, как выглядит орнамент кладки стены здания на фоне голубого неба или зеленой листвы, сравнить варианты орнамента ткани, выполненные в различной цветовой гамме, и т.п.

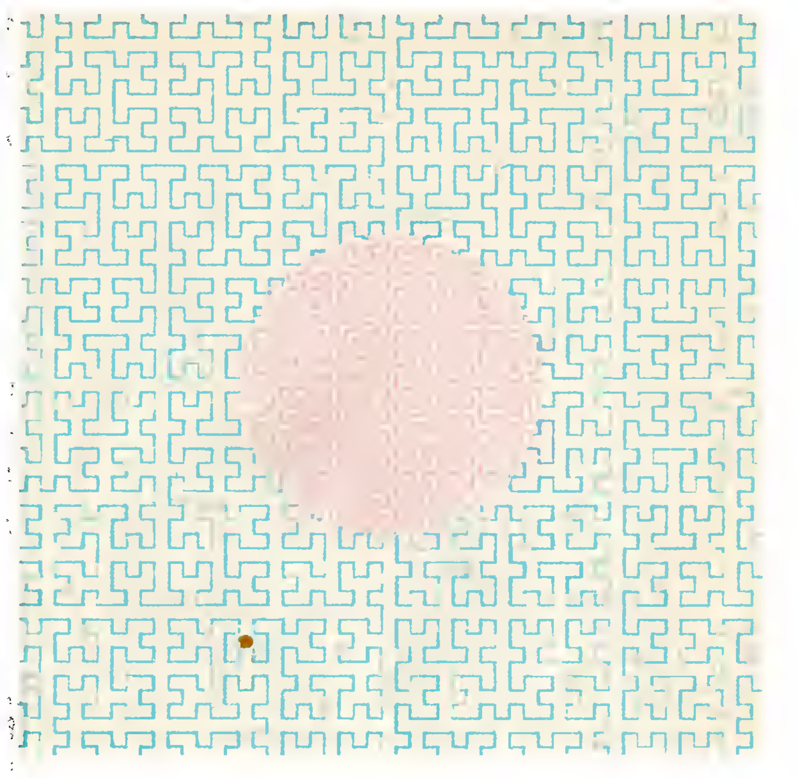


Рис. 63. Кривая Пеано

В современных графических системах высокого качества число генерируемых цветов может достигать нескольких миллионов !

Кроме автоматизации традиционных операций, компьютер открывает и совершенно новые возможности — орнамент можно создавать по математическим формулам. Математикам известно большое число функций (кривых), которые отличаются высокими эстетическими свойствами. Только с помощью тригонометрических функций синуса и косинуса можно создавать большое разнообразие рисунков. "Придуманы" даже специальные кривые, называемые "заполняющими пространство", например кривая Пеано (рис. 63).

С помощью компьютера можно создавать затейливые узоры, которые все время проявляются в новых конфигурациях. Используя графическое изображение

множеств Мандельброта, можно достичь богатства формы и красок. Причем пользователь может сам подбирать себе рисунок по вкусу, задавая компьютеру несколько числовых параметров (см., например, статью А.К. Дьюдни "Программы, генерирующие узоры для обоев" в журнале "В мире науки" № 11 за 1986 г).

Компьютер может стать *архивом* дизайнера и хранить целые коллекции орнаментов, как запущенных в производство, так и использовавшихся в прошлые века разными народами. Причем найти нужный орнамент в памяти компьютера можно гораздо легче и быстрее, чем в альбоме. А в случае необходимости можно быстро внести в него необходимые изменения.

Компьютеры все шире используются для *управления станками*. Если обеспечить совместимость компьютера, на котором конструируется орнамент, и компьютера, управляющего станком, можно тут же запустить орнамент в производство. Для этого достаточно перенести магнитный диск с записью орнамента в компьютер, управляющий станком.

Компьютер — подмастерье архитектора. Задумывались ли вы, какие задачи приходится решать архитектору? Он должен выбрать оптимальный вариант объемно-пространственной структуры сооружения, определить пропорции отдельных частей и ритм их чередования. Вот, например, как учили проектировать фасад здания архитекторов конца XVIII в. (1792 г.):

"Верхние окна с нижними должны быть одинаковой широты и расположены на одной линии. Над четвероугольными окнами делается вверху сводец. Простенки между окнами делаются наравне с широтою окон или несколько поболее; итак, если половина длины здания разделится на число футов широты окна, то частное число, отбросив остаток, покажет число окошек, которое для соразмерности и симметрии должно быть нечетное; если же оно будет четное, то надлежит сделать перемену в широте окошек и шафтов, дабы число окошек вышло не ровное".

Современным зодчим большим подспорьем могут быть системы автоматизированного проектирования.

Мы составили небольшую программу, чтобы продемонстрировать те возможности, которые открывает персональный компьютер перед архитектором. Программа рассчитывает и вычерчивает фасады домов. Мы ввели в нее данные о некоторых типах окон, дверей,

орнамента, присвоив каждому из них имя: ОКНО ГОЛЛАНДСКОЕ, ОКНО МОДЕРН, ОКНО СО СТАВНЯМИ, ДВЕРЬ ПРОСТАЯ, ОРНАМЕНТ "ПОД КИРПИЧ", и т.п. Архитектор задает число этажей и окон на этаже, выбирает тип окон и декоративное оформление фасада, задает форму и размеры окон и дверей. Работа с компьютером происходит в интерактивном режиме. Задав начальные значения входных параметров, архитектор запускает программу, которая вычерчивает на экране дисплея фасад здания. Первоначальный замысел можно откорректировать или идти дальше — заняться отделкой фасада.

Программа работает следующим образом.

1. Исходя из заданного числа этажей и окон, вычисляет координаты центров окон.

2. Вызывает подпрограмму построения указанного типа окна.

3. Строит окно указанного типа и размера в соответствии с заданием.

4. Вызывает подпрограмму отделки фасада. Затем определяет область, свободную от окон, и заполняет ее орнаментом, закрашивает.

На рис. 64 приведены примеры фасадов, построенных с помощью описанной программы. Заметим, что понятие "окно" здесь используется в более широком, чем обычно, смысле. Задавая "окно", мы задаем на экране дисплея область, в которой может быть вычерчено что угодно: балкон, лепка, ваза, колонна (рис. 65). При построении колонны, являющейся цилиндрическим телом, ее ребра — вертикальные линии — на проекции распределяются по закону

$$x = R\cos\,(i/n\cdot 360)\,,$$

где n — число ребер; i — номер вычерчиваемого ребра; x — координата i-го ребра.

Эти же соотношения используются и при вычерчивании кирпича или плитки на любых цилиндрических поверхностях.

Мы видим, что компьютер ничего не делает "на глазок". Вычерчиванию каждой линии предшествует точный расчет. Стоит компьютер один раз "обучить" некоторым расчетам, и он будет их выполнять автоматически, освобождая вашу "голову" от решения однообразных задач.

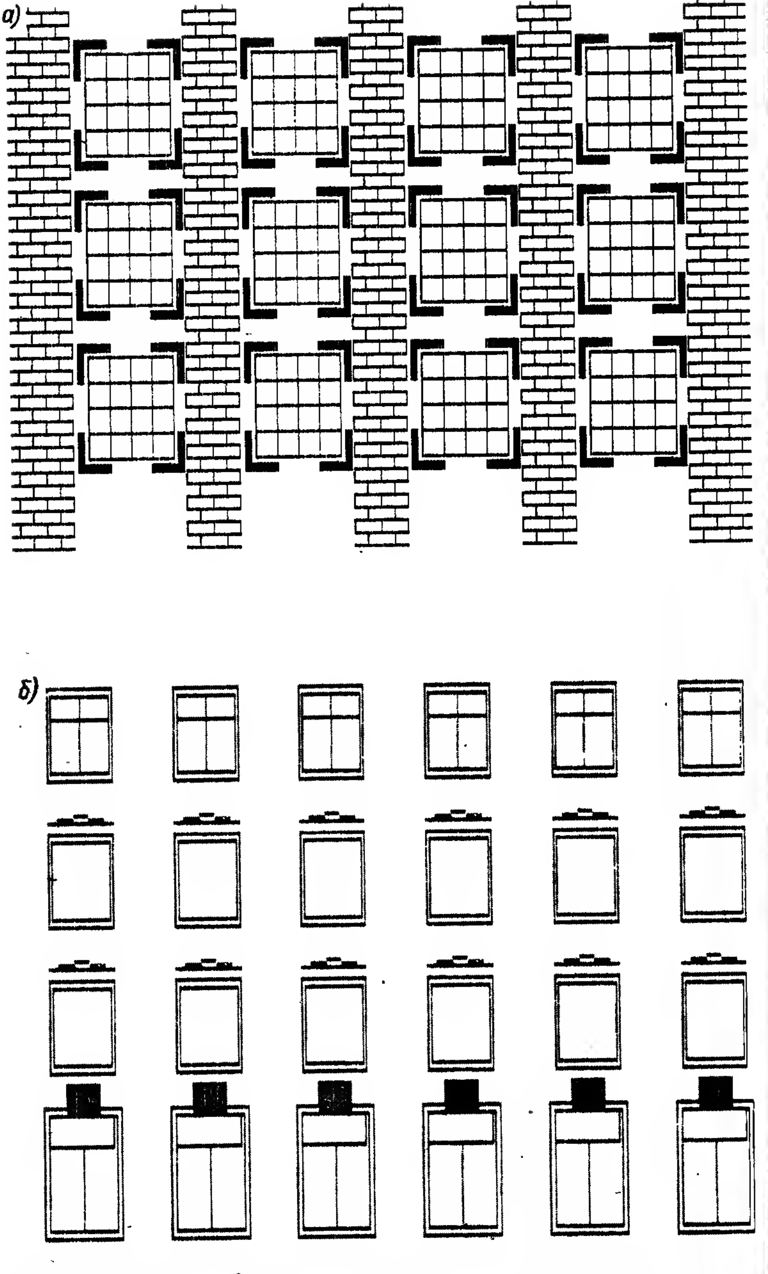
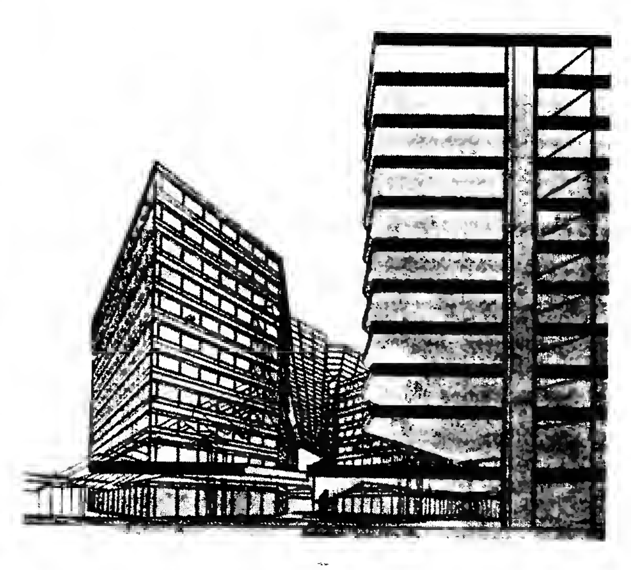


Рис. 64. Варианты фасадон, построенные с помощью компьютера

Рис. 65. Колоннада



ис. 66. Чертеж, полученный с помощью системы автоматизированного проектирования

В нашем примере фигурировала простая графическая система на базе персонального компьютера. Гораздо богаче возможности, представляемые архитектору большими графическими системами. Работая на АРМ, архитектор может создавать пространственную модель здания, начиная с наброска и кончая всеми требуемыми чертежами. Вместо изготовления макета здания моделируют его на экране дисплея (рис. 66). Скорость проектирования по сравнению с ручным проектированием увеличивается в 40—50 раз. Система хранит до 2000 стандартных элементов (несущих конструкций, окон, дверей, паркета и т.п.), вычерчивает архитектурные и инженерные чертежи всех видов, осуществляет пространственное планирование, проектирование интерьеров, подсчитывает стоимость проекта.

7. КОМПЬЮТЕРНОЕ ИСКУССТВО

Начало компьютерному искусству положили математики. Оказалось, что многие математические функции обладают высокими эстетическими свойствами и в то же время ввести их в компьютер и манипулировать ими очень удобно с помощью стандартной клавиатуры. Особенно удобными оказались функции sin и cos: получаемые изображения всегда размещаются в центре экрана, с их помощью можно получать большое разнообразие изображений за счет изменения небольшого числа переменных (рис. 67).

Случайное и детерминированное в искусстве. Другое направление компьютерного искусства заключается в развитии у компьютера "навыков" самостоятельного рисования. Компьютер создает композиции из геометрических фигур, форма, размер, положение, ориентация и цвет которых задаются не программистом, а случаем.

В 60-е гг. в прессе появились любопытные сообщения о том, что один предприимчивый художник для создания абстрактных произведений живописи использовал осла. Привязав к его хвосту кисть, он окунал ее в краску, после чего ставил осла задом к полотну. Осел размахивал хвостом, кисть оставляла на полосе разноцветные полосы. Как ни странно, но на такие "картины" находились покупатели. При использовании вычислительной техники можно было бы обойтись и без осла —

достаточно в компьютер ввести программу, описывающую траекторию помахивания хвостом и добавить к ней элемент случайности. В компьютере элемент случайности задается с помощью подпрограммы, называемой *генератором случайных чисел.* Генерируемые числа обычно лежат в пределах от нуля до единицы. Так, на рис. 68 координаты точек задаются последовательностью случайных чисел: значения, откладываемые по оси x, лежат в пределах от 0 до 320: 153, 264, 36, 9, 105, 123 ..., а по оси y — от 0 до 200: 12, 22, 180, 126, 68, 10 ... .

Изображение на рис. 68 совершенно случайно. Хаотическое расположение точек не несет никакой информации и не имеет эстетической ценности.

Введем в изображение элемент структуры (порядка) — окружности, сохранив при этом случайность выбора координат их центров (рис. 69). Это изображение, на наш взгляд, более привлекательно. Оно чем-то напоминает картину кипения воды.

Для построения рис. 70 также использовался генератор случайных чисел. (Художник-программист вывел на графопостроитель 65 000 точек, на это потребовалось 6 чП.) Здесь плотность распределения точек определялась еще и тремя концентрическими окружностями. Изображение имеет некоторую эстетическую ценность. Если бы оно было строго детерминированным (упорядоченным), т.е. состояло только из трех окружностей, то показалось бы примитивным.

На рис. 71 ”Скалистый берег” мы наблюдаем переход от полной упорядоченности до полной бесформенности, хаоса. Как видно из рисунка, и чистая синусоида и полный хаос неэстетичны. Наше внимание в основном задерживается на средней части рисунка.

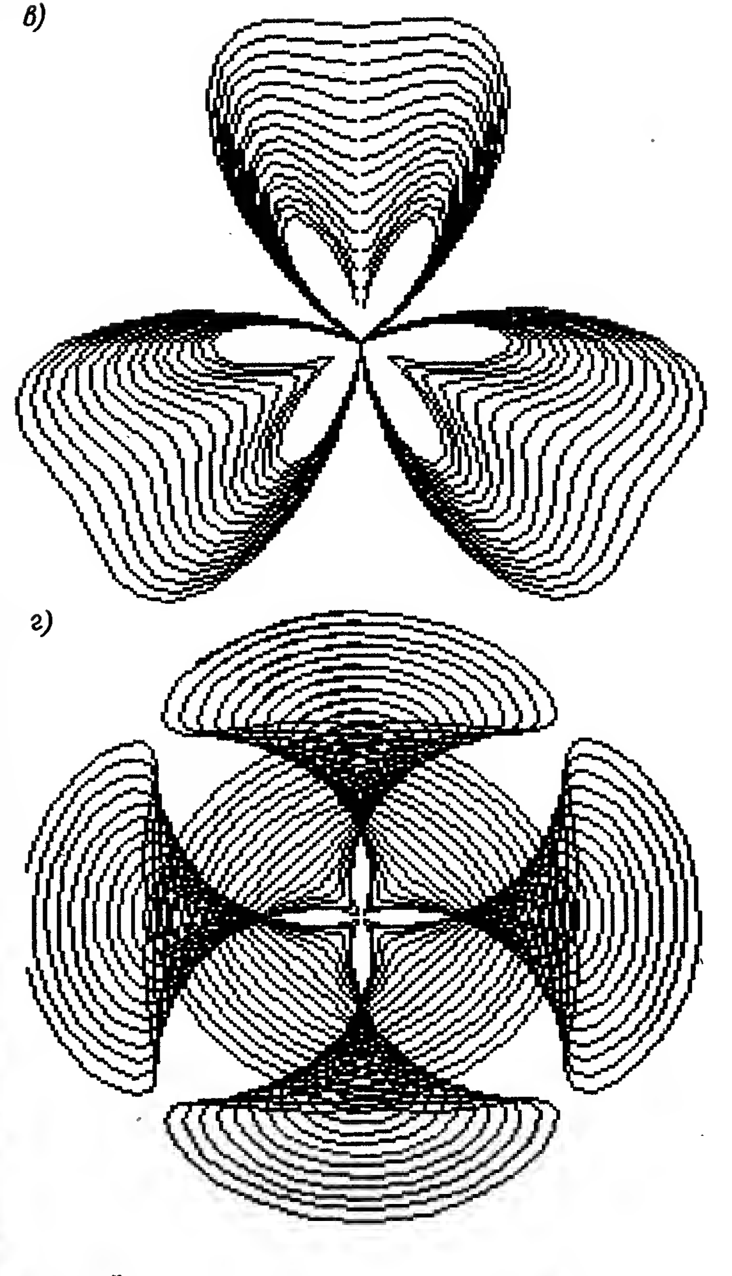
В музыке мы также наблюдаем сочетание упорядоченного и случайного. Так, Моцарт изобрел своеобразный способ сочинения мелодий: игральные кости выкидывались на специальную таблицу, определяющую выбор очередного такта создаваемой пьесы. Великие композиторы ясно сознавали роль случайности в общем строении музыкального сообщения.

Итак, компьютер предоставляет нам широкую гамму возможностей — от строгой упорядоченности до чистой случайности. Причем упорядоченность и случайность можно сочетать в любых пропорциях.

а)

б)

Рис. 67. Вариации на тему $r = A(1 + \cos^n k\varphi) +$



$B \sin^n k\varphi$, где r — радиус; A, k, φ, n, B — переменные

Рис. 68. Случайное расположение точек

Рис. 69. Окружности случайной природы

Рис. 70. Приближающиеся пределы

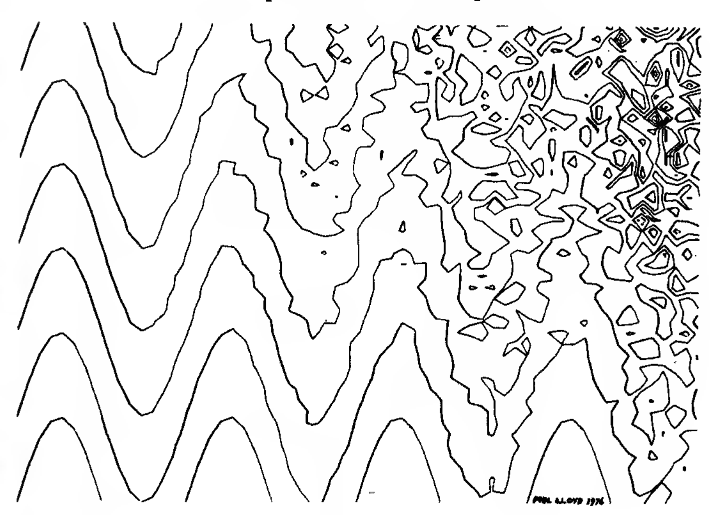


Рис. 71. Скалистый берег. Фил Ллойд. 1976

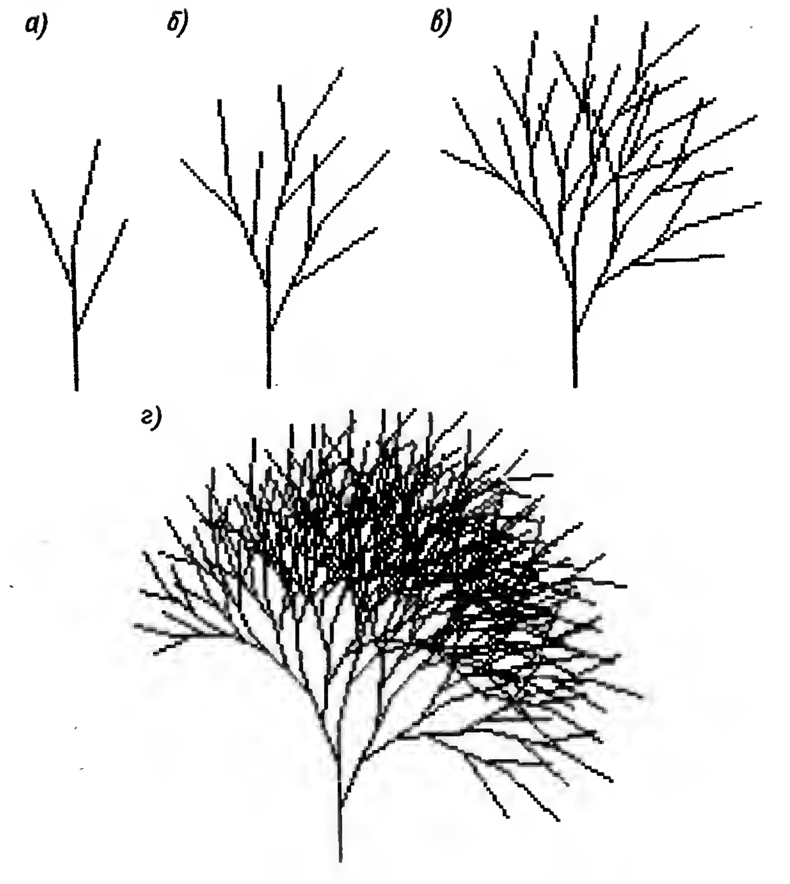


Рис. 72. Этапы построения дерева:
а — **первый шаг;** *б* — **второй шаг;** *в* — **третий шаг;** *г* — **четвертый шаг**

Рассмотрим следующий пример. У нас есть программа, которая позволяет строить изображение дерева. Программа моделирует "рост" дерева по принципу "из подобного растет подобное". Прежде всего пользователь задает параметры *изображения-эталона:* число ветвей, их длину и ориентацию, а также число шагов. Программа на каждом шаге повторяет эталон, т.е. из каждой ветки "вырастает" подобная ей ветка (рис. 72).

Изменение эталона приводит к построению деревьев различного типа. Так, на рис. 72 показаны этапы построения "березы", на рис. 73 — "елки", на рис. 74 —

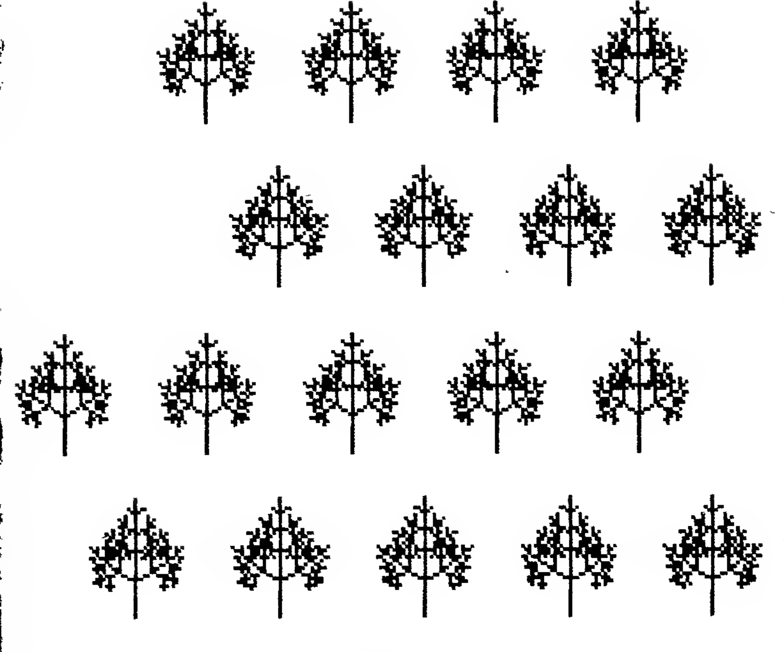


Рис. 73. Молодые посадки. Фиксированное расположение саженцев

"сосны". Из этих рисунков видно, что изменение только трех параметров эталона приводит к построению различного класса изображений.

На рис. 73 эталон повторяется через равные расстояния. Так выглядит лес, посаженный человеком. Если вы хотите, чтобы деревья располагались случайным образом, воспользуйтесь генератором случайных чисел. При каждом запуске программы вы получите совершенно новое изображение (рис. 74, 75).

Такие программы могут иметь прикладное значение, например: проектировщик дороги может "посадить" вдоль нее деревья, архитектор — создать "зеленый фон" для сооружения или ансамбля.

Вместо того, чтобы хранить в памяти компьютера большое количество образов деревьев различных пород, достаточно иметь всего лишь одну программу и варьировать начальные условия. Действительно, значение законов освобождает нашу память от запоминания массы подробностей.



Рис. 74. Молодой лес. Случайное расположение деревьев

Чем более программа учитывает законы развития леса, влияние различных факторов (солнца, ветра, силы тяжести, радиации), тем реалистичнее будет синтезируемое изображение.

Компьютер — архив изображений. Человек решил создать картину "Чайки на берегу моря". Рисовать он не умеет, но в его графическом архиве хранятся изображения чаек (рис. 76). Эти изображения он может вызвать из памяти компьютера, добавив для большей правдивости будущей картины изображение морской звезды. Итак, пользователь рисовать не умеет, но у него есть пять различных изображений чаек, одно морской звезды и фантазия. Он представляет себе композицию картины следующим образом: на берегу моря теснятся птицы, одни клюют молюсков, другие медленно идут, третьи просто стоят, наблюдают. Птицы различаются размерами и пространственной ориентацией.

Итак, замысел есть. Как его реализовать, не утруждая себя рисованием? Дадим имена имеющимся графическим изображениям, например: ПТИЦА 1, ПТИЦА 2,

Рис. 75. Лес. Случайное расположение деревьев

ПТИЦА 3, ... , ЗВЕЗДА. Теперь достаточно вызвать по имени каждую из птиц и задать для нее координаты x, y, определяющие положение левого нижнего угла изображения птицы. Чтобы добиться разнообразия в размерах птиц, можно использовать программу изменения масштаба, а с помощью программы зеркальных отображений — изменять их ориентацию. После того как координаты птиц и необходимые аффинные преобразования заданы, необходимо вызвать программу маскировки (стирания невидимых линий). Изображения, которые

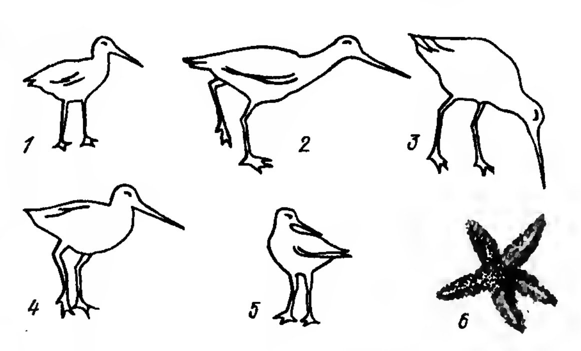


Рис. 76. Изображеиия, хранящиеся в архиве

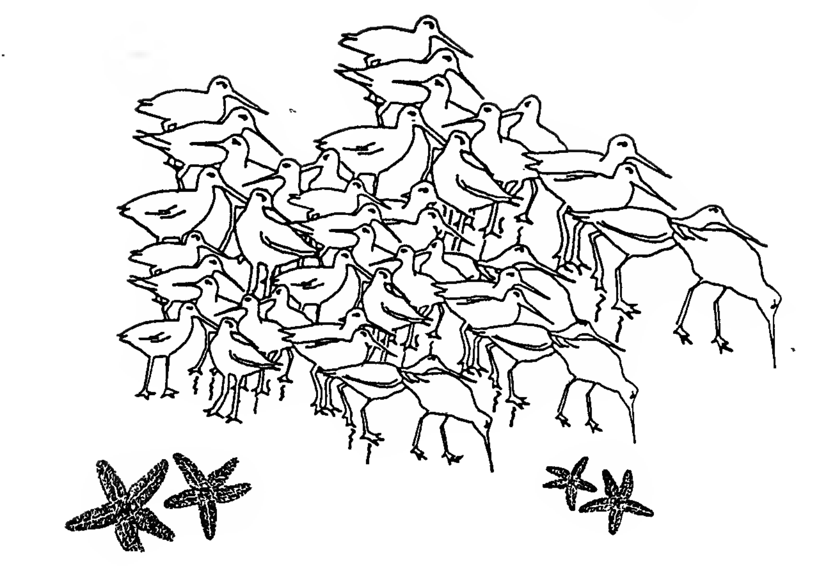


Рис. 77. Чайки

находятся ближе к наблюдателю, заслоняют те, что расположены выше по оси y, т.е. удаленные от наблюдателя. Программа маскировки "убирает" невидимые линии, создавая таким образом эффект глубины. Для того чтобы добиться наивысшей выразительности, пользователь после вывода изображения на дисплей может изменять расположение птиц, их ориентацию или размер. Таким путем был получен рис. 77.

Если мы его повнимательнее рассмотрим, то увидим, что программист значительно упростил себе работу создав из птиц небольшие группы. После этого он манипулировал не отдельными птицами, а целыми группами. Попробуйте их выявить.

Аналогичным образом были нарисованы изображения четырех морских звезд. Для их получения были применены операции вращения, масштабирования и сдвига.

Для создания картины можно выбрать и другой путь. Использовав генератор случайных чисел, можно написать программу, которая сама располагает птиц на берегу, задает их размеры и ориентацию. При каждом исполнении программа будет выдавать на экран изобра-

жение, не похожее на предыдущее. А эстетические вкусы пользователя могут выразиться только в отборе наиболее интересного с точки зрения его изображения. (При этом пользователь заранее не знает, что нарисует ЭВМ .) Причем чем точнее математическая модель описывает поведение чаек в сообществе, тем более реалистичным будет изображение. Но в то же время его не нужно уподоблять фотографии, поскольку главным достоинством картины остается степень ее эмоционального воздействия на зрителя.

Фотография копирует мгновенное состояние окружающего мира, а модель действительности, "заложенная" в компьютер, "копирует" законы ее развития. Поэтому картинка, отображаемая на дисплее, изображает то, что могло бы быть.

Электронная аппликация. Мы рассмотрели с вами, как работают с графической системой математики с художественным воображением и художники с математическими наклонностями. А как быть человеку, который ясно представляет себе замысел некоторой картины, но рисовать не умеет? Ведь он легко может оказаться в положении незадачливого волшебника, который "сделать хотел грозу, а получил козу, розовую козу с желтою полосой".

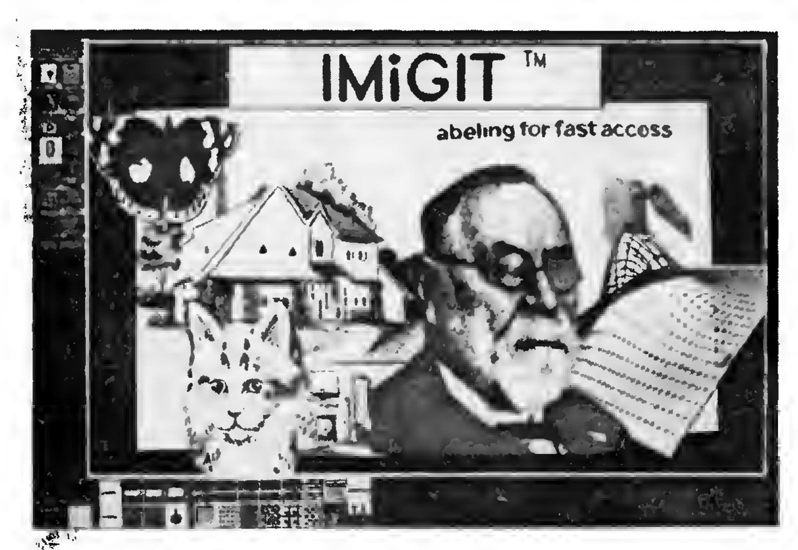


Рис. 78. Электронная аппликация

Представим, что у нас — пользователей персонального компьютера — имеется несколько сотен тоновых изображений, записанных на магнитном диске, а также набор программ — инструмент для работы с ними.

Изображения, которые занимают весь экран, называются *слайдами*. Каждый слайд состоит из четверти миллиона элементов изображения. Можно использовать как весь слайд, так и его *фрагмент*. Пользователь "вырезает" фрагмент путем перемещения и масштабирования особого фрагментирующего окна, так чтобы фрагмент попал в пределы окна. Фрагмент, подобно слайду, можно запомнить, а потом считать. Пользователь может создавать небольшие изображения одно за другим, модифицировать их по отдельности, а затем формировать композицию, электронный коллаж. Он может "вклеивать" заголовки, изменять цвета фрагментов. Пример электронной аппликации приведен на рис. 78.

Компьютерный вернисаж. А теперь мы предлагаем вам взглянуть на "картинки с выставки" (рис. 79—99). Речь идет о ежегодной выставке компьютерного искусства на которую представляют свои работы программисты из разных стран. Создание этих своеобразных композиций развивает фантазию, позволяет открыть новые интересные возможности. Кроме рисунков, взятых из журнала "Компьютер энд Пипл" (Computer and People, USA), на нашем вернисаже есть работы наших соотечественников (рис. 89, 96—99).

Попытаемся разобраться в принципах построения рисунков и впечатлениях, которые они вызывают.

Вы, конечно, заметили, что многие изображения беспредметны, представляют собой сочетание простых геометрических фигур, в то же время они производят интересный оптический эффект. Это произведения "оп-арта" (от англ. optical art — оптическое искусство).

Визуальный эффект достигается различными способами. Так, при пересечении линий под малыми углами возникает узор, называемый "муаром" (рис. 79, 80 и 89). Интересные эффекты, вызванные саккадическими (скачкообразными) движениями глаз, возникают при рассматривании рис. 81 и 84.

Рис. 96 иллюстрирует тот факт, что наш глаз даже в абстрактных пятнах пытается увидеть реальные объекты: мы четко видим на нем несуществующие белые квадраты.



Рис. 79. Цветы на Фортране. Томас Хьюстон. 1972

Рис. 80. Семейство эллипсов

Рис. 81. Излучение

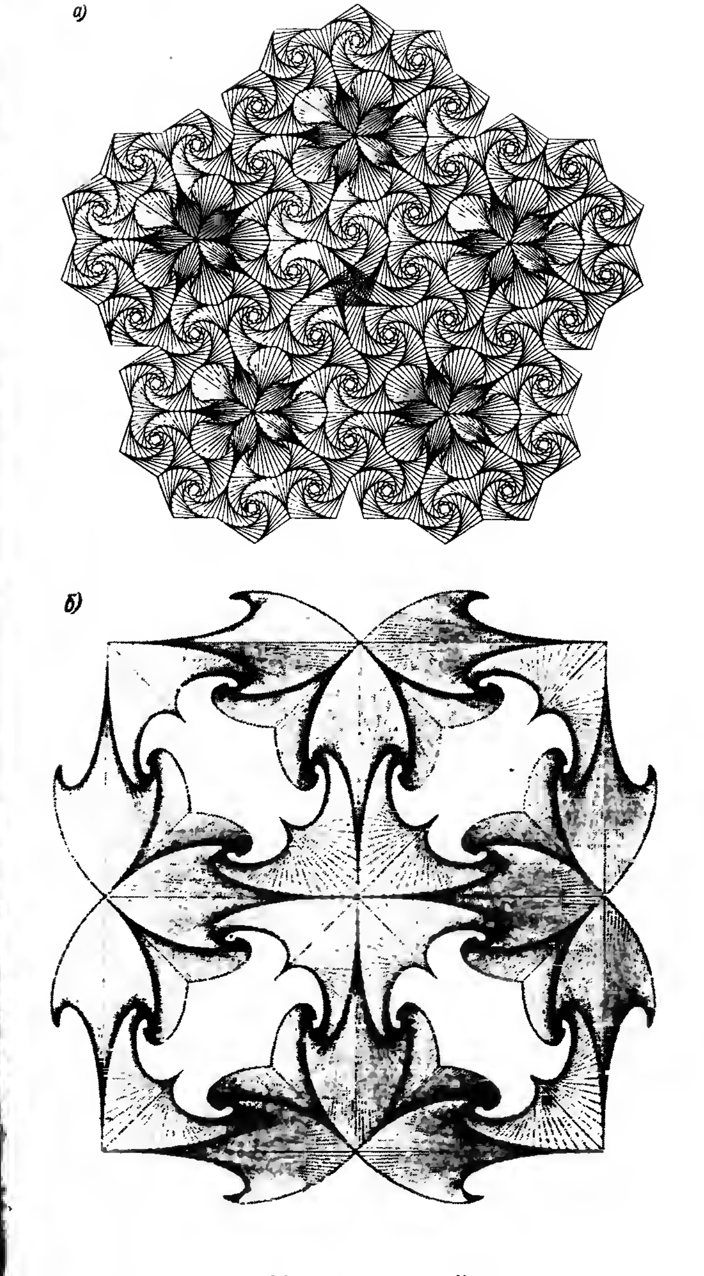


Рис. 82. "Математические" узоры



Рис. 83. Звери на поле

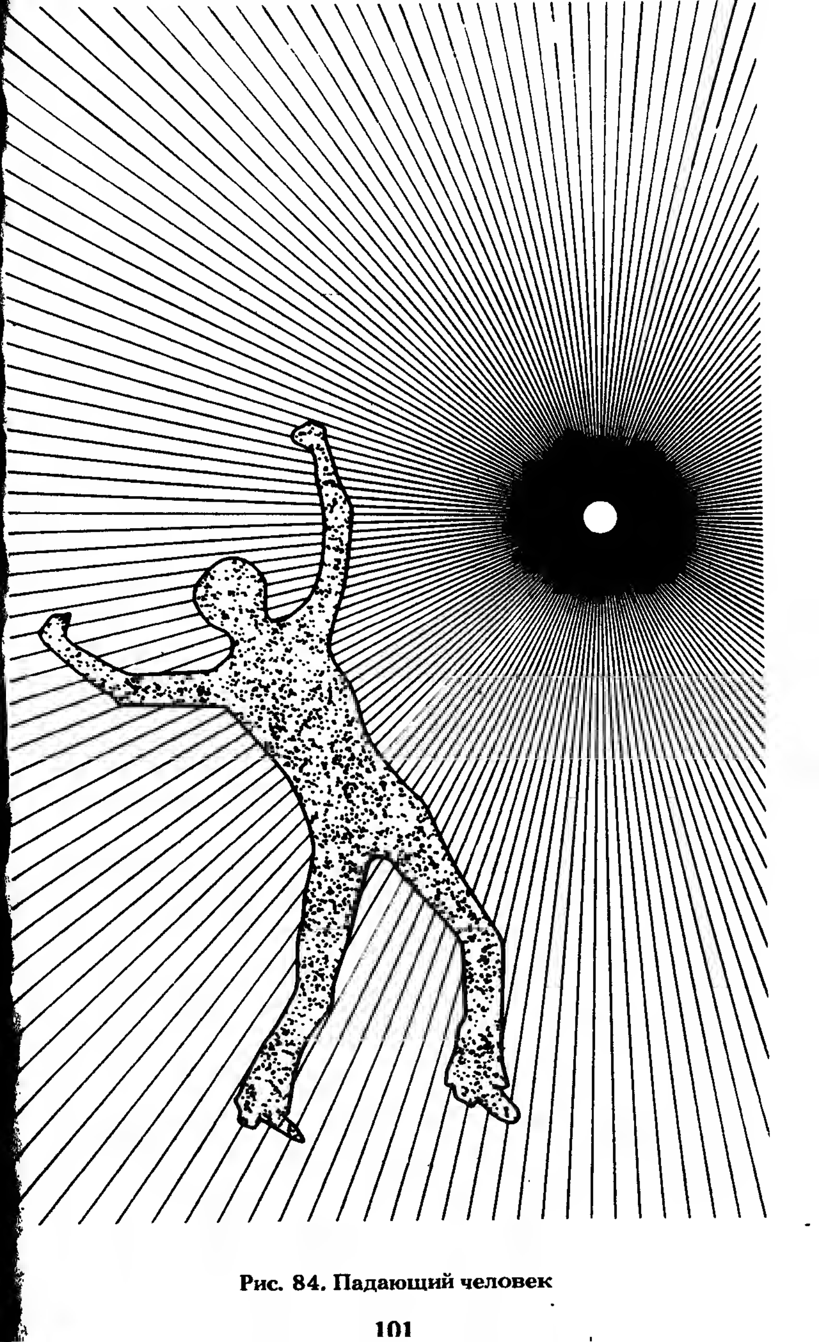


Рис. 84. Падающий человек



Рис. 85. Танец бабочек. Муцуко Сасаки. 1981

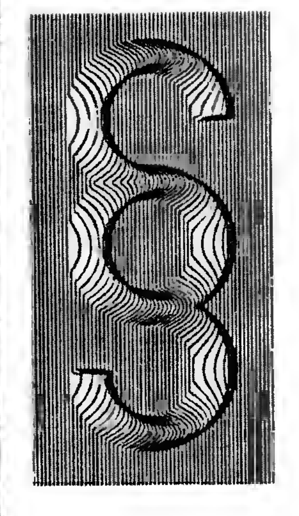


Рис. 86. Параграф



Рис. 87. Маска

Рис. 88. Черно-белые зигзаги

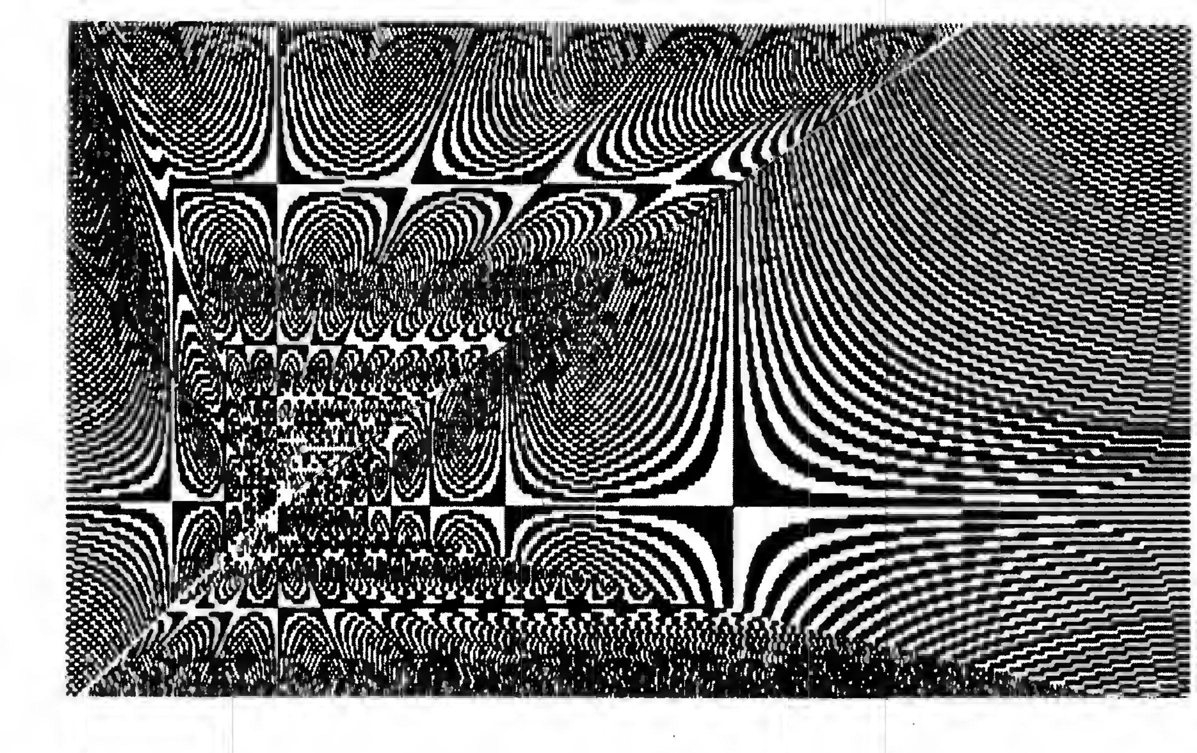


Рис. 89. Муар



Рис. 90. Утки в пруду. Муцуко Сасаки

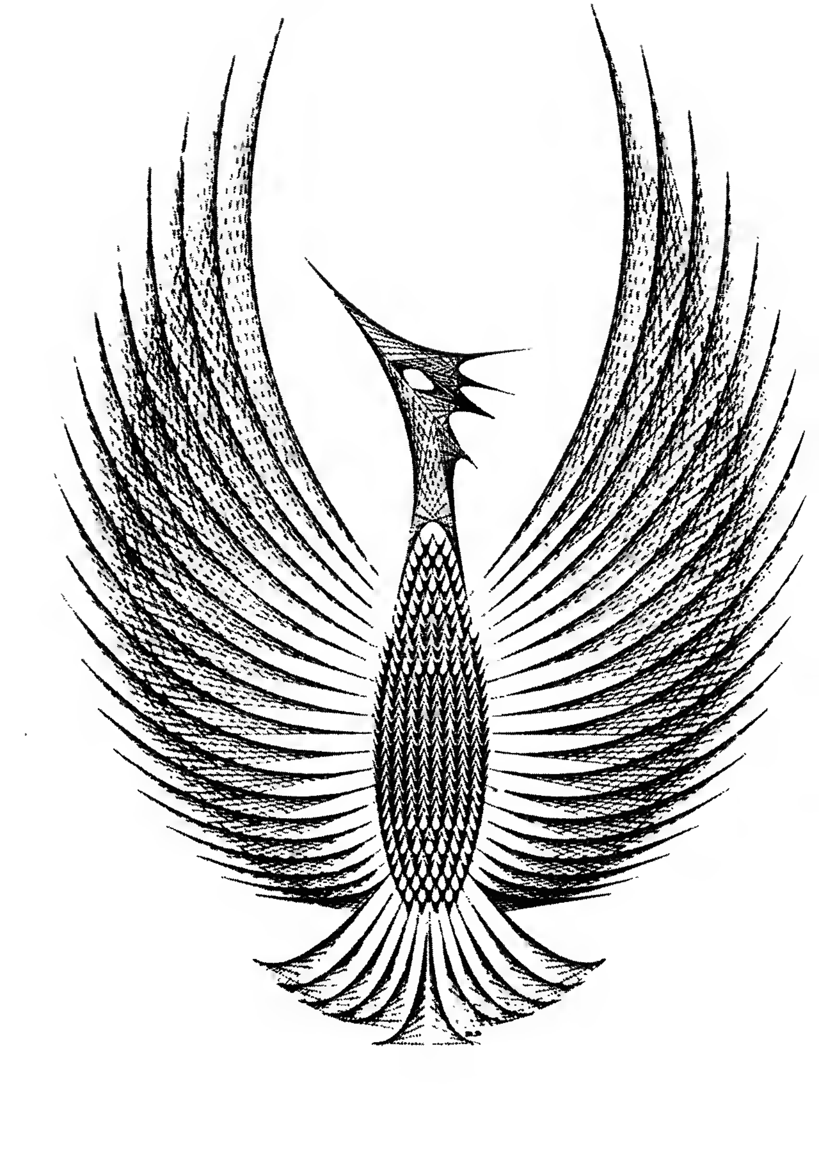


Рис. 91. Птица Феникс

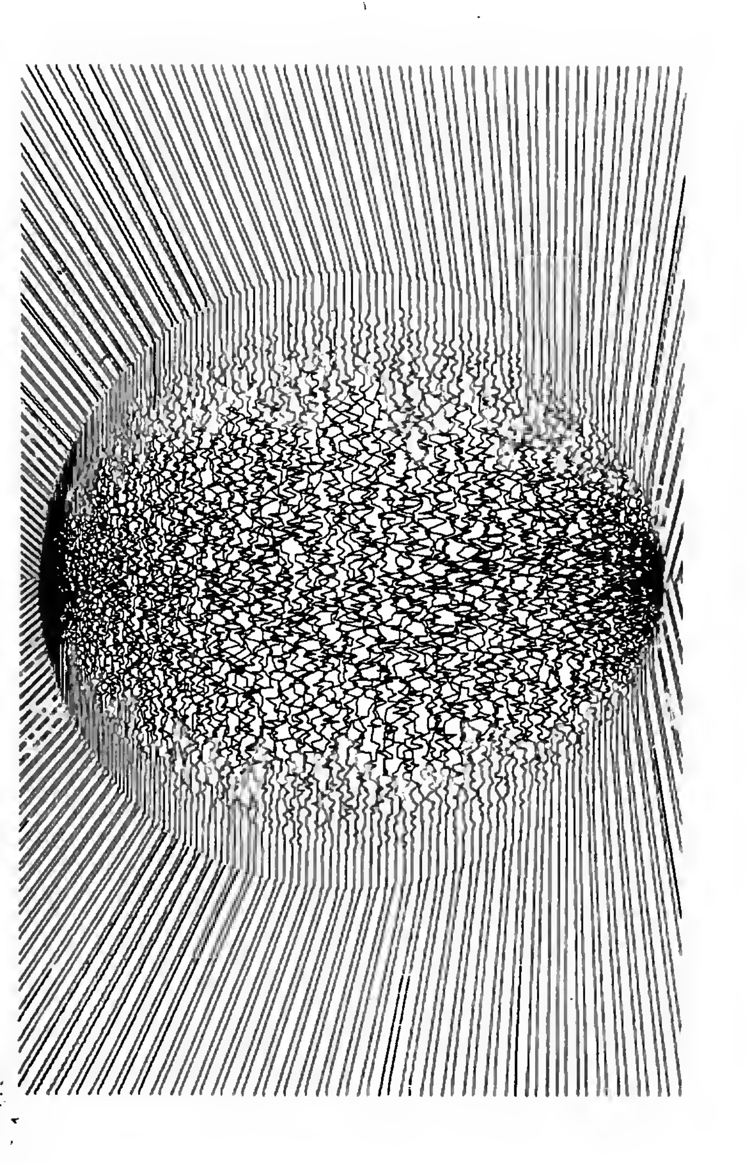


Рис. 92. Черная звезда

Рис. 93. Будущее человечества

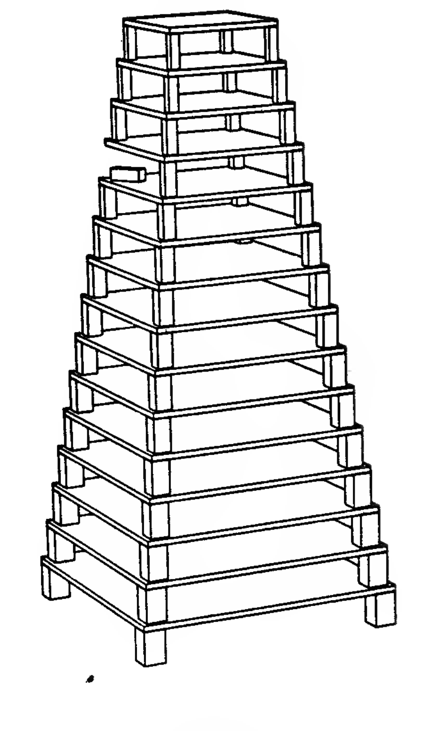


Рис. 94. Пирамида

Рис. 95. Радость катания на лыжах

Рис. 96. Несуществующие квадраты



Рис. 97. Кентавр

При рассматривании рис. 82, 87, 88, 91 сразу возникает вопрос: какие математические функции были здесь применены. К сожалению, авторы изображений эту информацию не публикуют.

При создании ряда работ использован генератор случайных чисел. Так, с его помощью на рис. 79 построены лепестки цветов, на рис. 84 "закрашена" фигура человека, а на рис. 92 вычерчены кривые внутри окружности.

Создатели рис. 83, 85 и 95 пользовались аффинными преобразованиями. В каждом рисунке при внимательном изучении можно выявить изображения-эталоны: животных и деревьев (рис. 83), лыжников (рис. 95), бабочек (рис. 85). Для создания композиции применены операции масштабирования, переноса, зеркального отображения.

Вы, очевидно, полагаете, что создание таких изображений под силу только программисту высокой квалификации. Но взгляните еще раз на рис. 96—99. Они построены по программам, написанным школьниками — победителями конкурса машинной графики журнала "Квант" (1981, № 9). Программы написаны на языке



Рис. 98. Львенок

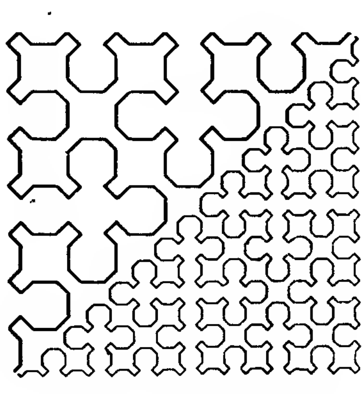


Рис. 99. Линии Серпинского

ПАСКАЛЬ. "Кентавра" придумала Света Безбородова из Рубцовска, "Линии Серпинского" — Сережа Баталов из Арзамаса, "Львенка" — Дима Корнилин из Горького. Так что не боги горшки обжигают!

От статики к динамике. Особенностью человеческого восприятия является обостренная реакция на новое, ведь обычно именно новое несет в себе информацию. Поскольку изображения в стиле оп-арта кроме визуального эффекта ничего не несут, подогревать интерес к ним можно за счет "оживления", т.е. перехода от статичных изображений к динамичным.

Действительно, мы можем часами, как завороженные смотреть на пламя свечи или костер, на игру солнечных бликов на воде, но едва ли будем дольше минуты рассматривать их фотографии.

Тенденция расширения изобразительных возможностей за счет введения движения наблюдается в скульптуре. Появляются так называемые "мобили" — конструкции с подвижными элементами. Художников динамических (изменяющихся во времени) изображений называют "кинетами" (от "кинетики").

И здесь заманчивые перспективы открывает компьютер. На нем можно моделировать не только абстрактные предметы, но и огонь, горящий в камине, радугу и плывущие облака, фейерверк, молнию и многое другое.

Всем вам хорошо известен калейдоскоп. Эта игрушка, в сущности, генерирует изображения в стиле оп-арта. Каждое изображение случайно, оно определяется расположением цветных стеклышек в поле зрения, но структура изображения детерминирована; за счет многократного отражения от зеркал одной простейшей комбинации создается симметричное изображение. Прелесть калейдоскопа — в непредсказуемом изменении картинки при его вращении. Не представляет особого труда промоделировать калейдоскоп на персональном компьютере с цветным дисплеем. Роль цветных стеклышек выполняют точки на экране, высвеченные с помощью генератора случайных чисел. Зеркальные отражения можно моделировать с помощью простых математических формул.

Итак, подведем итоги. Что же дает графическая система художнику или просто человеку, обладающему фантазией?

Во-первых, это *инструмент для рисования.* В вашем распоряжении линии любой толщины, начертания, цвета, и интенсивности, цветовые плоскости, окрашиваемые в любой из миллиона цветов.

Во-вторых, это *инструмент для манипуляций с изображением и вычислений.* Построение перспективного изображения объекта в любом ракурсе; светотени, блики, полутоны; вращение, масштабирование, перемещение, зеркальное отображение — вот диапазон возможностей компьютера.

В-третьих, это *хранилище изображений.* Компьютер хранит и оперативно воспроизводит контурные, тоновые и цветные изображения одежды, мебели, утвари, оружия разных исторических эпох, животных и растений и т.д. (Правда, в полной мере эти возможности могут быть реализованы только при увеличении внешней памяти за счет перехода на запоминающие устройства на оптических дисках.

Диалог энтузиаста и скептика. Каким же будет художник будущего, например 2000 года? Мы попытались взглянуть на этот вопрос с двух точек зрения — энтузиаста электронно-вычислительной техники и скептика. И вот что у нас получилось.

Энтузиаст: "Художник, родившийся в 2000 г., будет совсем другим. Ему не надо знать, как держать кисть, разводить краски, грунтовать полотно. Ни к чему знание особенностей графики, акварельной техники, живописи маслом. У него есть компьютер, который все знает и умеет.

Компьютеру известны законы отображения и преломления света. Для него не представляет особого труда добиться высокой реальности объектов — изобразить прозрачность стекла, блеск металла, матовость бархата, воздушную дымку и даже небо над Сатурном.

Художник будущего — программист-жонглер. Являясь абонентом информационно-вычислительной сети, он "вытаскивает" изображения-заготовки из любой библиотеки и легко манипулирует ими. У него прекрасная память, он хорошо знает графическое программное обеспечение.

А главное, у него свое видение мира, безудержная фантазия, для реализации которой имеются все технические и программные средства".

Скептик: "При рождении фотографии многие полагали, что живопись обречена. Прошло время. Живопись

осталась, а фотография с трудом пробивает себе дорогу в искусство.

При рождении кино предрекали смерть театру. Прошло время. Театр живет, а в некоторые театры даже не попасть — столько желающих.

Такое же брожение умов наблюдалось при появлении телевидения. Но телевизор также не вытесняет театр и кино, а дополняет их.

То же можно сказать и о компьютерном искусстве. Оно не заменит традиционные графику и живопись, а лишь дополнит их, займет свое место в искусстве, видимо, в основном в декоративно-прикладных областях, например в дизайне".

8. КИНОСТУДИЯ "КОМПЬЮТЕРФИЛЬМ"

> Мультипликация является единственным подлинным искусством в настоящее время, потому что в ней, и только в ней, художник абсолютно свободен в своей фантазии и может делать все, что угодно.
>
> *Ч. Чаплин*

Вы, конечно, знаете, что за ярким, жизнерадостным зрелищем — мультфильмом — стоит долгий, однообразный и утомительный труд десятков художников. Ведь в мультфильме движение создается ими, оно не существует в реальной действительности.

И в этой области применение машинной графики приносит удивительные плоды. Достаточно сказать, что при переходе от традиционных форм к использованию графических систем продолжительность создания мультфильма сокращается более чем в 2 раза!

Движение. Если два дискретных события сменяют друг друга быстрее, чем через 1/16 с, мы воспринимаем их как одновременные, они сливаются. С другой стороны, если события развертываются слишком медленно, а именно с промежутками свыше 8—10 с, ощущение движения исчезает, вместо него мы воспримем только ряд отдельных состояний объекта.

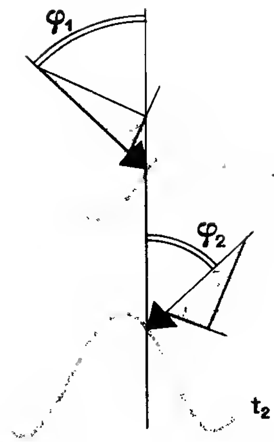


Рис. 100. Качка на море

Возьмем для примера движение стрелок часов. В то время как секундная стрелка перемещается, "скачет" у нас на глазах, две другие стрелки неподвижны, застыли в одном положении. Перемещение минутной и часовой стрелки мы зрительно не воспринимаем, хотя и понимаем, что они движутся.

Итак, если мы хотим изобразить движение, то отдельные состояния должны сменяться не чаще, чем 1 раз в 1/16 с и не реже чем 1 раз в 8—10 с. В современном кино частота смены кадров составляет 24 кадра/с.

Предположим, вы — художник-мультипликатор и хотите нарисовать яхту, летящую "в волнах на раздутых парусах" (рис. 100). Задайте значения максимальных углов крена, а также продолжительность периода качки T. Компьютер высчитывает необходимое число кадров $n = 24T$ и для каждого кадра (момента времени) вычисляет угол крена. После этого им вычерчивается новое положение яхты и волны. Каждое изображение, генерируемое на экране дисплея, вы снимаете на кинопленку, одно изображение — один кадр. А вот другая задача. Дано:

Мотор! Хлопушка! Выстрел! Кровавая луна!
И тут я в титрах выпадаю из окна.

Высота окна над уровнем земли — H.
Требуется определить необходимое число кадров n.

Положение тела в каждом кадре, т.е. на каждую i-ю 1/24 часть секунды, определим по формуле

$$S_i = H \frac{1/24\ ig}{2}$$

Время падения, т.е. продолжительность сцены, равно $T = \sqrt{2H/g}$. Необходимое число кадров $n = 24T$.

Одной из самых сложных и трудоемких в мультипликации является работа, обратная озвучиванию: тексту, который произносит персонаж, должна соответствовать артикуляция (движения губ, щек, подбородка). Каждой фонеме (звуку речи) соответствует своя








Рис. 101. Говорит призрак

артикуляция. Нам намного легче понимать речь говорящего, если мы смотрим на него, а глухонемые могут "читать" по губам. Если закономерности артикуляции выразить в математической форме и ввести в компьютер, то можно для каждой фонемы обеспечить соответствующую "деформацию", мимическую картину.

На рис. 101 вы видите созданную художником модель лица, которую "оживил" программист. Чтобы создать иллюзию объемности, изображение сглажено и "подсвечено". Программа может управлять движением глаз и бровей. Режиссер с клавитатуры вводит текст, который должен быть произнесен (или пропет) персонажем, задав дополнительно темп речи, продолжительность акцентируемых фонем, мимику лица, изменение цвета лица, пространственного положения говорящего (повороты, наклоны) и т.п. Все остальное, как говорится, дело техники.

Снимается кино[1]. Как же складывается работа съемочной группы с графической системой?

---

[1]Этот материал взят из журнала "ТИИЭР", 1984, т.72, № 1, с. 30—42.

Получив сценарий, художник просматривает сцены и изображает ключевые кадры будущего фильма так, как он их себе представляет. Режиссер-постановщик выделяет объекты, составляющие сцену, будь то космический корабль, дом или гуманоид. По типу архитектурного чертежа составляется подробное описание объектов, которое несет в себе полную трехмерную информацию о каждом из них.

В описании поверхность каждого объекта представляется как совокупность многоугольников. Чем сложнее объект, тем больше многоугольников необходимо для описания его формы, следовательно, реалистичное воспроизведение таких объектов требует больших объемов памяти ЭВМ и затрат машинного времени.

Далее составленные рисунки кодируются. Координаты точек на чертеже преобразуются в числа и вводятся в память компьютера. Используя полученные данные, он может построить на экране дисплея каркасное изображение трехмерной модели. Сложный комплекс аппаратных и программных средств позволяет поворачивать объект на экране дисплея, рассматривать его и вносить необходимые поправки.

На следующем этапе каркасному (проволочному) изображению придается реалистический вид с помощью специального математического обеспечения, позволяющего имитировать поверхности из различных материалов: металла, стекла, дерева, бархата, пластика, мела. Поверхность можно сделать блестящей или матовой, прозрачной или полупрозрачной. Можно показать световые блики и тени.

Работая с текстом предварительного просмотра, режиссер по техническим видам съемок имеет возможность указать компьютеру действия, которые должны произойти в промежутках между ключевыми кадрами, задать время показа отрывка. На заключительном этапе создания фильма ЭВМ последовательно рассчитывает кадр за кадром, полностью используя программы по формированию облика объектов.

К объектам, участвующим в сцене, применяются команды по их перемещению, уже отработанные на "проволочных" изображениях. В зависимости от сложности на создание кадра может уйти от 10 до 400 с. Результаты расчета поступают на записывающее устройство. Получившиеся кусочки ленты монтируют с фрагментами, отснятыми с натуры.

Объем данных, воспроизводимых на экране дисплея, равен произведению числа элементов изображения (1280×1024) на 24 бита. Всего для одного кадра необходимо приблизительно 4 Мбайт данных.

Цвет каждого элемента изображения задается тремя числами, соответствующими основным цветам — красному, зеленому и синему. В 24-битовом слове, выделяемом на один элемент, для каждого из этих чисел отводится по 8 бит. Для качественной записи на 35-мм пленку необходимо, чтобы на каждый из 3000×4000 элементов изображения приходилось 30 бит, по 10 бит на каждый из трех цветов. Всего для одного цветного кадра необходимо 45 Мбайт. Пусть для определения одной составляющей цвета у каждого элемента изображения требуется выполнить 10 операций над числами с плавающей запятой. Поскольку фильм показывают со скоростью 24 кадра/с, то на получение фрагмента, который будет длиться 1 с, понадобится произвести 8,64 млрд операций.

В действительности различные алгоритмы освещения и изображения поверхности могут потребовать от 1 до 10 тыс. операций на каждую цветовую составляющую. Поэтому на получение односекундного мультфильма уходит от 864 млн до 864 млрд арифметических операций. Для суперкомпьютера, способного выполнять в секунду 200 млн операций, для генерации односекундного мультфильма необходимо от 3 с до 10 ч. Правда, вовсе не обязательно менять в каждом кадре все изображение целиком, поэтому при грамотном программировании объем вычислений можно значительно сократить.

Если средняя степень сложности кадра составляет 250 тыс. многоугольников, то с помощью суперкомпьютера в месяц можно производить 12-минутный фильм, в год — 144-минутный. Это эквивалентно по длительности двум полнометражным художественным фильмам. При использовании же традиционного метода Диснея на каждый двухчасовой фильм в среднем уходит по 2 года.

В традиционном кинематографе кинооператор и художник по специальным эффектам вынуждены перемещаться с камерой перед построенными миниатюрными моделями, постоянно сталкиваясь с механическими ограничениями. Все происходит в реальном мире, подчиняющемся физическим законам, в том числе закону гравитации. В то же время при моделировании на ЭВМ художники и режиссеры нисколько не привязаны к реальности и получают возможность создавать свои миры. "Объектив" может быть направлен из любой точки трехмерного пространства. Объекты могут запросто игнорировать гравитацию, выворачиваться наизнанку, проходить через твердые тела, летать по сложным траекториям, изменять свою фор-

му. Они могут разрываться на куски, которые тут же будут собираться воедино.

Через несколько лет компьютер можно будет использовать не только для получения необычных эффектов, он станет основным инструментом режиссера. Режиссер будет описывать объект, характер его взаимодействия с другими объектами, а машина уже сама сможет реалистично "оживить"сцены и объекты. Для каждой ситуации будет автоматически определяться и реалистично воспроизводиться алгоритмическое поведение объекта.

Тем, кто заинтересовался машинной графикой и хочет углубить свои познания, мы рекомендуем почитать следующие книги:

Ньюмен У., Спрулл Р.С. Основы интерактивной машинной графики/Пер. с англ. М.: Мир, 1976, 573 с.

В книге описаны принципы построения дисплеев различных типов, необходимые для работы в интерактивном режиме технические средства и программное обеспечение. Большое внимание уделено работе с изображениями пространственных объектов, проектированию графических систем.

Фоли Дж., ван Дэм А. Основы интерактивной машинной графики/Пер. с англ. В 2-х книгах. М.: Мир. 1985. I кн.: 365 с. II кн.: 368 с.

Книга I посвящена векторной графике. Рассмотрены устройства ввода и вывода графической информации, методы организации графического диалога. Приведено более 40 цветных иллюстраций, полученных на графических системах.

Книга II содержит данные о современных векторных и растровых дисплеях. Приведены алгоритмы растровой развертки и закраски, удаления скрытых линий и поверхностей, а также модели освещения и методы описания цвета.

Гилой В. Интерактивная машинная графика/Пер. с англ. М.: Мир, 1981. 380 с.

В книге рассмотрены методы преобразования изображений, графические языки, технические и программные средства реализации графических систем.

Институт прикладной математики, начиная с 1970 г., выпускает информационные издания, посвященные описанию графических программ, входящих в состав комплекса ГРАФОР, под редакцией Ю.М. Баяковского.

ПРИЛОЖЕНИЕ

Ниже приведены изображения, полученные с помощью графической системы (рис. П1—П6) и программы к ним.



Рис. П1. Улица

Программа к рис. П1

```
155 CI10 CLS
15 COLOR 5,4
20 RANDOMIZE 17
30 FOR K=0 TO 7
40 D=INT(RND*3+1)
50 FOR I=1 TO D
60 Y=I*26
70 X=K*36+10
80 LINE(X,154-Y)-(X+36,180-Y),,B
```

```
90 FOR L=1 TO 7
100 X1=4*L
110 IF L<4 OR L>5 THEN LINE(X+X1,170-Y)-
  (X+4+X1,170-Y-6),,B ELSE GOTO 120
120 NEXT L
130 NEXT I
140 LINE(X,154-Y)- (X+18,154-Y-15)
145 LINE-(X+18,154-Y-18)
150 LINE(X+18,154-Y-15)-(X+36,154-Y)
155 CIRCLE(X+18,154-Y-7),3
160 NEXT K
170 CIRCLE(50,40),25,2
175 PAINT(50,40),2
180 FOR D=0 TO 360 STEP 45
185 DRAW"ta=d;bu25nu10bd25"
187 NEXT D
190 DRAW"bm10,182nr300"
200 TRAILOR$="u12r24d12n124r4u8r4nf4d4r4d418"
210 DRAW"bm40,180;xtrailor $;"
220 CIRCLE(44,180),2
230 CIRCLE(60,180),2
240 CIRCLE(74,180),2
250 COLOR 1,2
```



Рис. П2. Спираль

Программа к рис. П2

```
10 CLS
15 COLOR 0,2
20 PI=3.14159
30 NV=15 ´kol—vo vitkov
40 NC=3 ´......zahodov
50 S=.15 ´step ygla
60 BETA=2*PI/NC
70 DR=S*110/(NV*PI*2)
80 FOR C=1 TO NC ´zahod
90 R=0:PSET(160,100),1
100 FOR V=1 TO NV ´vitki
110 FOR FI=0 TO 2*PI STEP S
120 R=R+DR
130 X=R*COS(FI+BETA*(C—1))+160
140 Y=.9*R*SIN(FI+BETA*(C—1))+100
150 LINE—(X,Y),C
160 NEXT:NEXT:NEXT
```



Рис. П3. Цветок

Программа к рис. П3

```
10 CLS
11 COLOR 1,2
15 PI=3.14159
20 FOR B=10 TO 50 STEP 10
```

```
25 A=B/2
40 FOR D=0 TO 360 STEP 4
50 FI=2*PI*D/360
60 R=A*(1+COS(5*FI))+B*SIN(5*FI)^2
70 X=R*COS(FI)+160
80 Y=R*SIN(FI)+100
85 IF D=0 THEN LINE(X,Y)—(X+1,Y+1)ELSE LINE—(X,Y),1
90 NEXT
100 NEXT
110 PAINT(4,104),1
```



Рис. П4. Куб

Программа к рис. П4

```
10 CLS
12 COLOR 1,0
15 A=20
16 C=0
30 FOR Y=20 TO 160 STEP 30
35 C=C+1
36 IF C=INT(C/2)*2 THEN D=50 ELSE D=20
40 FOR X=D TO 220 STEP 40
50 LINE(X,Y)—(X+A,Y+A),2,BF
55 LINE(X,Y)—(X+A,Y),1
60 LINE(X,Y)—(X+A/2,Y—A/2),1
```

```
70 LINE-(X+A*1.5,Y-A/2),1
80 LINE-(X+A*1.5,Y+A/2),1
82 LINE-(X+A,Y+A),1
85 LINE(X+A*1.5,Y-A/2)-(X+A,Y),1
90 PAINT(X+A/2,Y-A*.1),1
100 NEXT:NEXT
```

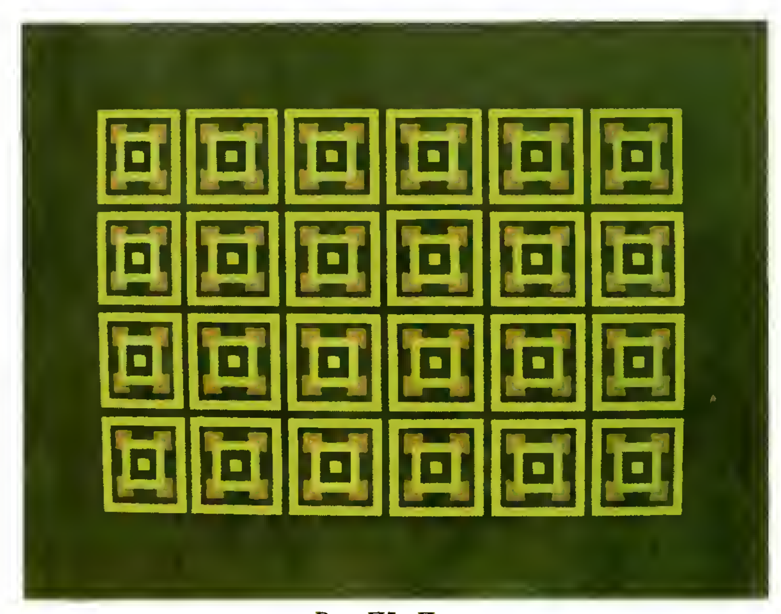


Рис. П5. Паркет

Программа к рис. П5

```
5 A=22:B=11:D=4:C=4
10 CLS
15 COLOR 0,4
20 FOR Y=25 TO 180 STEP 49
25 FOR X=40 TO 310 STEP 49
27 LINE(X-A,Y-A)-(X+A,Y+A),3,BF
29 LINE(X-A+D,Y-A+D)-(X+A-D,Y+A-D),0,BF
32 LINE(X-B-D,Y-B-D)-(X-B+D,Y-B+D),2,BF
34 LINE(X-B-D,Y+B-D)-(X-B+D,Y+B+D),2,BF
36 LINE(X+B-D,Y-B-D)-(X+B+D,Y-B+D),2,BF
38 LINE(X+B-D,Y+B-D)-(X+B+D,Y+B+D),2,BF
60 LINE(X-B,Y-B)-(X+B,Y+B),3,BF
65 LINE(X-B+D,Y-B+D)-(X+B-D,Y+B-D),0,BF
67 LINE(X-D*.5,Y-D*.5)-(X+D*.5,Y+D*.5),,BF
70 NEXT:NEXT
```



Рис. П6. Солнце

Программа к рис. П6

```
5 CLS
7 SCREEN 1
8 COLOR 1,2
10 PI=3.14159
20 LINE(3,1)—(317,199),,B
30 LINE(6,3)—(314,197),,B
35 K=20
37 LINE(3+K,11)—(317—K,187),,B
40 LINE(6+K,13)—(314—K,185),,B
50 LINE(8+K,15)—(312—K,184),1,BF
55 N=310/10
115 'COLOR 0,1
120 CIRCLE(70,80),20,2
130 PAINT(70,70),2
150 FOR D1=—3.14 TO 3.14 STEP .3
151 D=D1+3.14
152 R=ABS(D1)*3.5
154 X1=20*COS(D)+70
155 Y1=20*SIN(D)+80
160 X2=(20+R^2)*COS(D)+70
161 Y2=(20+R^2)*SIN(D)+80
166 LINE(X1,Y1)—(X2,Y2),2
```

```
170 NEXT
180 FOR I-1 TO 9
190 A=I*15
200 LINE(160+A, 15)-(90+A, 160), 2, , &H9999
220 NEXT I
225 FOR J=0 TO 20 STEP 5
230 FOR I=1 TO 8.7*PI-J/19 STEP .5
235 IF I=1 THEN C=0 ELSE C=1
240 X=I*10+18+J/2
250 Y=10*SIN(I)+150+J
255 IF I=1 THEN LINE(X, Y)-(X+1, Y),0 ELSE LINE-(X,Y), 2
260 NEXT: NEXT
270 LINE(205,150)-(202,140), 2
280 LINE-(215,145), 2
290 LINE-(205,150), 2
295 PAINT(210,145), 2
300 LINE (208,145)-(228,100),2
310 LINE-(227,145), 2
320 LINE(235,147)-(205,138), 2
330 PAINT(220,131), 2
```

ОГЛАВЛЕНИЕ

НАУЧНО-ПОПУЛЯРНОЕ ИЗДАНИЕ

Александров Виктор Васильевич,
Шнейдеров Виталий Семенович

РИСУНОК, ЧЕРТЕЖ, КАРТИНА НА ЭВМ

Редактор Т.Г. Филатова
Художественный редактор С.С. Венедиктов
Технический редактор Т.П. Малашкина
Корректор И.Г. Иванова
Обложка художника В.В. Пожидаева

ИБ № 5266

Сдано в набор 26.01.87. Подписано в печать 02.09.87. М—18471. Формат $84\times108^{1}/_{32}$. Бумага офсетная № 1. Гарнитура Сенчури. Печать офсетная. Усл. печ. л. 6,72. Усл. кр.-отт. 35,7. Уч.-изд. л. 6,27. Дополнительный тираж 70 000 экз. Заказ 3984. Цена 80 к.

Ленинградское отделение ордена Трудового Красного Знамени издательства "Машиностроение", 191065, Ленинград, ул. Дзержинского, 10

Предприятие малообъемной книги дважды ордена Трудового Красного Знамени Ленинградского производственного объединения «Типография имени Ивана Федорова» Союзполиграфпрома при Государственном комитете СССР по делам издательств, полиграфии и книжной торговли. 192007, Ленинград, ул. Боровая, 51, с оригинала-макета, изготовленного в издательстве "Машиностроение" на наборно-пишущих машинах

(ЛИНИЯ ОТРЕЗА)

Информатика, компьютер, ЭВМ помогут Вам в учебе, игре, работе и в повседневной жизни

УВАЖАЕМЫЙ ЧИТАТЕЛЬ!

Издательство просит Вас в прилагаемой анкете отметить позиции, соответствующие Вашей оценке этой книги.

1. Появилось ли у Вас желание иметь личный персональный компьютер?

Да Нет

2. Какой процент новой информации почерпнули Вы из книги?

100% 75? 50% 25% 0

3. Ваша оценка содержания книги.

Отлично Хорошо Удовлетворительно

4. Помогла ли Вам книга в повседневной работе или учебе?

Да Нет

5. Какие по Вашему мнению нужны научно-популярные книги по компьютерной технике?

____

____

____

Фамилия . . . .

Имя . . . .

Отчество . . . .

Возраст . . . .

Образование . . . .

Кем работаете (где учитесь)? . . . .

. . . .

Кем хотели бы работать? . . . .

. . . .

. . . .

Просим отрезать страницу по линии отреза и в почтовом конверте выслать по адресу:
191065 Ленинград, ул. Дзержинского, 10, ЛО издательства "Машиностроение".

В.В. Александров, В.С. Шнейдеров. Рисунок, чертеж, картина на ЭВМ

(ЛИНИЯ ОТРЕЗА)

0 к.

Рисунок, чертеж, картина на ЭВМ

Компьютер вторгается во все сферы человеческой деятельности. Превратившись из большого калькулятора в интеллектуальный инструмент, он изменил лицо тысячи профессий. Миллионы людей садятся за рабочие места, оснащенные средствами вычислительной техники. Компьютер вычерчивает шрифты, строит проекции деталей, вращает и перемещает их, изменяет масштаб, генерирует цветные и тоновые изображения различных поверхностей. Компьютер становится помощником дизайнера, архитектора, художника-мультипликатора. Он позволяет сделать их труд более производительным и творческим. О технических устройствах, входящих в состав графических систем, особенностях работы с ними и возможностях их применения рассказывает эта книга.


АЛЕКСАНДРОВ, В.С. ШНЕЙДЕРОВ
Рисунок
чертеж
картина
на ЭВМ